



# **NAVAL POSTGRADUATE SCHOOL**

**MONTEREY, CALIFORNIA**

## **THESIS**

**A PERSONAL INERTIAL-NAVIGATION SYSTEM  
BASED ON MULTIPLE DISTRIBUTED,  
NINE-DEGREES-OF-FREEDOM, INERTIAL  
MEASUREMENT UNITS**

by

Cole C. Johnson

December 2016

Thesis Advisor:  
Second Reader:

Xiaoping Yun  
James Calusdian

**Approved for public release. Distribution is unlimited.**

THIS PAGE INTENTIONALLY LEFT BLANK

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
<b>1. AGENCY USE ONLY</b> (Leave blank)		<b>2. REPORT DATE</b> December 2016		<b>3. REPORT TYPE AND DATES COVERED</b> Master's thesis
<b>4. TITLE AND SUBTITLE</b> A PERSONAL INERTIAL-NAVIGATION SYSTEM BASED ON MULTIPLE DISTRIBUTED, NINE-DEGREES-OF-FREEDOM, INERTIAL MEASUREMENT UNITS			<b>5. FUNDING NUMBERS</b>	
<b>6. AUTHOR(S)</b> Cole C. Johnson				
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Naval Postgraduate School Monterey, CA 93943-5000			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>	
<b>9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> N/A			<b>10. SPONSORING / MONITORING AGENCY REPORT NUMBER</b>	
<b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____ N/A ____.				
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release. Distribution is unlimited.			<b>12b. DISTRIBUTION CODE</b>	
<b>13. ABSTRACT (maximum 200 words)</b>  The use of inertial-measurement units (IMUs) for personal navigation is investigated in this thesis. IMUs lack a position-finding algorithm that optimally blends sensor data to achieve high accuracy in a GPS-denied environment. In this research, software and a methodology for tracking position using body-mounted IMUs, building on a gait-phase detection algorithm and quaternion-based complementary filter developed at the Naval Postgraduate School, is developed. The performance of a consumer-grade nine-degrees-of-freedom IMU is characterized and alternative sensor placements evaluated to determine optimal mounting location or locations. Measurements were fused from gyroscope, accelerometer, and magnetometer sensors to create a single, virtual IMU. In addition, measurements from a distributed system of IMUs, as well as multiple co-located IMUs, were averaged to find performance enhancements. Software was developed to streamline and integrate position solutions into a larger network of capabilities. Results show that the foot is the optimal mounting location, and other placements degrade performance. Averaging measurements from multiple IMUs at one location improves performance but with diminishing returns as the number of IMUs increase. We recommend that multiple IMUs be printed on the same MEMS circuit board to achieve accuracy by fusing the measurements of co-located sensors.				
<b>14. SUBJECT TERMS</b> personal navigation, inertial sensor, gyroscope, magnetometer, accelerometer, attitude heading and reference system, quaternion algorithm, inertial measurement unit, complementary filter, gait phase detection, zero velocity update, MEMS, IMU, AHRS, GPS denied, distributed sensor, virtual sensor, fusion, network-centric warfare, navigation warfare, electronic warfare, jamming, Reticle			<b>15. NUMBER OF PAGES</b> 167	
			<b>16. PRICE CODE</b>	
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified		<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified		<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified
<b>20. LIMITATION OF ABSTRACT</b> UU				

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release. Distribution is unlimited.**

**A PERSONAL INERTIAL-NAVIGATION SYSTEM BASED ON MULTIPLE  
DISTRIBUTED, NINE-DEGREES-OF-FREEDOM, INERTIAL MEASUREMENT  
UNITS**

Cole C. Johnson  
Civilian, Department of the Air Force  
B.S., Northern Arizona University, 2009

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**

from the

**NAVAL POSTGRADUATE SCHOOL  
December 2016**

Approved by: Xiaoping Yun  
Thesis Advisor

James Calusdian  
Second Reader

R. Clark Robertson  
Chair, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

## ABSTRACT

The use of inertial-measurement units (IMUs) for personal navigation is investigated in this thesis. IMUs lack a position-finding algorithm that optimally blends sensor data to achieve high accuracy in a GPS-denied environment. In this research, software and a methodology for tracking position using body-mounted IMUs, building on a gait-phase detection algorithm and quaternion-based complementary filter developed at the Naval Postgraduate School, is developed. The performance of a consumer-grade nine-degrees-of-freedom IMU is characterized and alternative sensor placements evaluated to determine optimal mounting location or locations. Measurements were fused from gyroscope, accelerometer, and magnetometer sensors to create a single, virtual IMU. In addition, measurements from a distributed system of IMUs, as well as multiple co-located IMUs, were averaged to find performance enhancements. Software was developed to streamline and integrate position solutions into a larger network of capabilities. Results show that the foot is the optimal mounting location, and other placements degrade performance. Averaging measurements from multiple IMUs at one location improves performance but with diminishing returns as the number of IMUs increase. We recommend that multiple IMUs be printed on the same MEMS circuit board to achieve accuracy by fusing the measurements of co-located sensors.

THIS PAGE INTENTIONALLY LEFT BLANK

## TABLE OF CONTENTS

<b>I.</b>	<b>INTRODUCTION.....</b>	<b>1</b>
<b>A.</b>	<b>IMPORTANCE OF POSITION.....</b>	<b>1</b>
1.	How GPS Works .....	1
2.	Benefits of IMUs.....	2
3.	Improving IMU Performance.....	2
<b>B.</b>	<b>RETICLE.....</b>	<b>3</b>
1.	The Geometry-of-Fires Problem .....	3
2.	Reticle Subprojects .....	4
<b>C.</b>	<b>PREVIOUS WORK.....</b>	<b>5</b>
<b>D.</b>	<b>THESIS OBJECTIVE .....</b>	<b>6</b>
<b>II.</b>	<b>BACKGROUND .....</b>	<b>7</b>
<b>A.</b>	<b>A BRIEF HISTORY OF POSITION FINDING.....</b>	<b>7</b>
1.	Longitude .....	7
2.	The Harrison Sea Clock .....	8
3.	Early Mechanical Spinning Gyroscopes .....	8
4.	Laser-Light Gyroscopes .....	10
5.	GPS.....	12
6.	MEMS .....	13
<b>B.</b>	<b>IMU CHARACTERISTICS .....</b>	<b>14</b>
<b>C.</b>	<b>IMU/ SENSORS USED IN THIS RESEARCH .....</b>	<b>15</b>
<b>D.</b>	<b>GYROSCOPE ERROR SOURCES.....</b>	<b>19</b>
1.	Error Bias Stability .....	20
2.	Scale-Factor Error (Non-Linearity and Asymmetry) .....	20
3.	Angular Random Walk .....	21
4.	Other Gyroscope Errors.....	22
5.	Gyroscopes in IMUs.....	23
<b>E.</b>	<b>HIGH-ACCURACY POSITION USING LOW-ACCURACY SENSORS .....</b>	<b>26</b>
1.	MEMS IMU Size Comparison .....	26
2.	Personal Navigation System.....	27
3.	Zero-Velocity Updates .....	28
<b>III.</b>	<b>EXPERIMENTS .....</b>	<b>33</b>
<b>A.</b>	<b>THE CHARACTERIZATION OF SENSOR ERRORS.....</b>	<b>33</b>
1.	Test Setup .....	33
2.	PSD of Sensor Outputs .....	36

3.	Histogram of Sensors .....	45
B.	THE COMPLEMENTARY FILTER .....	49
1.	Reference Frames.....	49
2.	Factored Quaternion Algorithm: Stance Phase .....	50
3.	Angular Rate Measurements: Swing Phase .....	53
4.	Gait-Phase Detection .....	54
C.	TUNING.....	58
1.	Athletic Track: Walking Only .....	58
2.	PNS Settings Explained .....	65
3.	PNS Performance when Walking .....	67
D.	PLACEMENT .....	77
1.	The Shin .....	79
2.	The Lower Back .....	82
3.	The Chest .....	86
4.	Conclusion .....	88
E.	DISTRIBUTED SENSORS.....	89
1.	Selection of Reference Frame: Body or NED .....	91
2.	Optimal Mounting Locations for Gyroscope, Accelerometer, and Magnetometer .....	93
F.	AVERAGING OF MULTIPLE FOOT-MOUNTED SENSORS .....	105
1.	Sample-Rate Inconsistencies .....	106
2.	Averaging Two Sensors .....	110
3.	Averaging Three Sensors .....	112
4.	Averaging Four Sensors .....	113
5.	Comparison of Additional Averaged Sensors .....	114
6.	Conclusion: More Sensors Enable Better Performance.....	118
IV.	CONCLUSION .....	119
A.	OPTIMAL MOUNTING LOCATION: THE FOOT.....	119
1.	Advantages.....	119
2.	Disadvantages.....	119
B.	DISTRIBUTED SYSTEM OF IMUS NOT ADVISED.....	120
C.	AVERAGING MULTIPLE, CO-LOCATED IMUS IMPROVES PERFORMANCE.....	121
D.	DEVELOPMENT FORECAST .....	121
	APPENDIX A: PARTIAL DATA ANALYSIS .....	123
A.	IMPORT LARGE YEI 3-SPACE TEXT FILES INTO MATLAB.....	123
B.	PSD OF ONE SENSOR.....	124

C.	PSDS OF ALL SENSORS.....	127
D.	HISTOGRAM PLOT OF SENSOR.....	130
E.	HISTOGRAM FUNCTION.....	131
APPENDIX B: DATA COLLECTION .....		135
LIST OF REFERENCES .....		137
INITIAL DISTRIBUTION LIST .....		145

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF FIGURES

Figure 1.	LTV-N-2 Loon Rocket Guidance System .....	9
Figure 2.	YEI 3-Space Data-Logging v2.0 AHRS .....	16
Figure 3.	YEI 3-Space Sensor Software Suite v3.0r7 .....	22
Figure 4.	IMU Form Comparison From Left to Right: YEI 3-Space Module, 3DM-GX1 (first generation), and 3DM-GX4-25 (fourth generation) .....	26
Figure 5.	Reticle GUI Home Screen. Adapted from [48]. .....	28
Figure 6.	Sphere Calibration Wizard before (Left) and after (Right) Accelerometer and Magnetometer Calibration .....	35
Figure 7.	Simulated PSD of Randomly Generated Noise (Top), and Same Data Processed using Welch PSD (Bottom) .....	38
Figure 8.	PSD of Sensor 1's Raw Triaxial Gyroscope Outputs .....	39
Figure 9.	PSD of All Four IMUs' Raw Triaxial Gyroscope Outputs .....	41
Figure 10.	PSD of Sensor 1's Raw Triaxial Accelerometer Outputs .....	42
Figure 11.	PSD of All Four IMUs' Raw Triaxial Accelerometer Outputs .....	43
Figure 12.	PSD of Sensor 1's Raw Triaxial Magnetometer Outputs .....	44
Figure 13.	PSD of All Four IMUs' Raw Triaxial Magnetometer Outputs .....	45
Figure 14.	Histogram of Sensor 1's Raw Triaxial Gyroscope Outputs .....	46
Figure 15.	Histogram with Two of Sensor 1's Raw Accelerometer Outputs .....	47
Figure 16.	Histogram with Two of Sensor 1's Magnetometer Outputs .....	48
Figure 17.	Combined Angular Rate, Sensor 3, Lap 9, Three Footsteps of Gait-Phase Detection, Data Not Squared, SNR Low .....	55
Figure 18.	Squared Combined Angular Rate, Sensor 3, Lap 9, Three Footsteps of Gait-Phase Detection, Data Squared, SNR High .....	57
Figure 19.	Athletic Track Used for Testing. Adapted from [64]. .....	59
Figure 20.	2×1 Sensor Brackets, Left-Foot Bracket (Left) and Right-Foot Bracket (Right) .....	61
Figure 21.	Right-Foot 2×1 Bracket Holding Two IMUs .....	61
Figure 22.	2×2 Sensor Bracket Holding Four IMUs in Fixed Relationship .....	62
Figure 23.	2×2 Bracket Mounted with Four IMUs on Right Foot .....	63
Figure 24.	Step-Plot of 3-Space Sensor Using MicroStrain 3DM-GX1 PNS Settings; Sensor 3, Lap 6 .....	68

Figure 25.	Step-Plot of 3-Space Sensor using Optimized PNS Settings; Sensor 3, Lap 6 .....	69
Figure 26.	Sensor 3's Step-Plots for Laps 4–6 using 2×2 Bracket and Settings Optimized for Lap 6.....	70
Figure 27.	Sensor 3's Step-Plots for Laps 1–3 Using 2×1 Bracket and Settings Optimized for Lap 6; Gains Not Tuned for Each Lap .....	71
Figure 28.	All Four Sensors Step-Plots for Lap 6 Using 2×2 Bracket and Settings Optimized for Sensor 3's Lap 6; Gains Not Tuned for Each Sensor.....	72
Figure 29.	Sensor 3's Step-Plots for Laps 7–9; Gains Tuned for Each Lap .....	74
Figure 30.	Sensor 3's Step-Plots for Laps 7–9; Gains Tuned for Sensor 4's, Lap 2.....	75
Figure 31.	Sensor 4's Step-Plots for Laps 7–9; Calibration Occurred at Lower Temperature; Settings Optimized for Lap 8 .....	76
Figure 32.	Distributed IMU Mounting Arrangement; Two per Foot, One per Shin, One on Lower Back, and One on Chest, for Eight Total.....	78
Figure 33.	Left-Shin Mounted IMU's Step-Plot .....	80
Figure 34.	Combined Angular-Rate Length for Left Shin-Mounted IMU, Lap 10.....	82
Figure 35.	Lower-Back Mounted Step-Plot for Sensor 2, Lap 9; Scale Factor 12.137; Each Footstep (Left and Right) was Detected .....	83
Figure 36.	Combined Angular Rate Length for Lower-Back Mounted Sensor 2, Lap 9 .....	85
Figure 37.	Chest Mounted Step-Plot for Sensor 2, Lap 8; Scale Factor 18.824; Each Footstep (Left and Right) was Detected .....	87
Figure 38.	Combined Angular-Rate Length for Chest-Mounted Sensor 3, Lap 8 .....	88
Figure 39.	2×1 Bracket Holding Sensors 1 and 2 on Lower Back.....	91
Figure 40.	GUI to Convert Body Frame Data to NED Coordinate System .....	92
Figure 41.	Step-Plot Displaying Baseline Reference using Sensor 3's Foot-Mounted Measurements from Lap 9; Scale Factor 1.01 .....	93
Figure 42.	Step-Plot with Unadjusted Scale and Direction, Lap 9; Foot-Mounted Gyroscope, Accelerometer/Magnetometer Back; Scale Factor 0.0.....	94
Figure 43.	Step-Plot with Scale Adjusted, Lap 9; Foot-Mounted Gyroscope, Accelerometer/Magnetometer Lower Back; Scale Factor 7.9 .....	95

Figure 44.	Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Gyroscope on Foot and Accelerometer/Magnetometer on Lower Back .....	96
Figure 45.	Step-Plot with Foot-Mounted Accelerometer, Gyroscope and Magnetometer from Lower Back; Lap 9, Scale Factor $-2.54$ .....	97
Figure 46.	Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Accelerometer on Foot and Gyroscope/Magnetometer on Lower Back .....	98
Figure 47.	Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Magnetometer on Foot and Gyroscope/Accelerometer on Lower Back .....	100
Figure 48.	Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Gyroscope/Accelerometer on Foot and Magnetometer on Lower Back .....	101
Figure 49.	2×1 Module Bracket with Sensors 1 and 2, Mounted on Lower Back ....	102
Figure 50.	2×1 Bracket with Sensors 3 and 4, Mounted on Right Foot .....	102
Figure 51.	Averaging Two Distributed Sets of IMUs in Poor Mounting Locations versus One IMU in Same Locations .....	103
Figure 52.	Averaging Two Distributed Sets of IMUs in Poor Mounting Locations versus One IMU in Same Locations .....	104
Figure 53.	Orientation of Sensors 1–4.....	106
Figure 54.	Misaligned Angular-Rate Length Data of Four Time-Aligned IMUs .....	107
Figure 55.	Same Angular Rate Length Data as in Figure 54, Except Measurements Manually Aligned .....	108
Figure 56.	IMU Sample Frequencies Displaying Inconsistencies Despite Identical Settings.....	109
Figure 57.	GUI to Fuse and Average Multiple Sensors into One Output .....	111
Figure 58.	Step-Plot Comparing Single Sensor versus Two Averaged Sensors .....	112
Figure 59.	Step-Plot Comparing One Sensor versus Three Averaged Sensors.....	113
Figure 60.	Step-Plot Comparing One Sensor versus Four Averaged Sensors .....	114
Figure 61.	Step-Plot Comparing Increasing Numbers of Averaged Sensors .....	115
Figure 62.	Line-plot Comparing Increasing Numbers of Averaged Sensors .....	115
Figure 63.	Zoomed-in Step-Plot: Start Versus End Position of Increasing Averaged Sensors; Same Data as Figures 61 and 62.....	116

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TABLES

Table 1.	Gyroscope Performance of Competing IMUs .....	24
Table 2.	PNS Settings, Optimized for MicroStrain 3DM-GX1. Source: [11]. .....	65
Table 3.	PNS Algorithm Values, Optimized for 3-Space Sensor .....	68
Table 4.	PNS Settings Optimized for Sensors 3 and 4 in Laps 7–9 .....	76
Table 5.	Optimal Settings for Left-Shin Mounted IMU, Lap 10 .....	81
Table 6.	Optimal Settings for Lower Back Mounted Sensor 2, Lap 9 .....	84
Table 7.	Optimal Settings for Chest Mounted Sensor 2, Lap 8 .....	87
Table 8.	Settings Used to Optimize Performance from Averaged Sensors .....	117
Table 9.	Data Collection Events .....	135

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF ACRONYMS AND ABBREVIATIONS

~	about
°/s	degrees per second
2D	two-dimensional space
3D	three-dimensional space
AHRS	attitude heading and reference system
ARW	angular random walk
BPM	beats per minute
C/N <sub>0</sub>	carrier-to-noise ratio
CAD	computer-aided design
cm	centimeter
dB	decibel
dBW	decibel Watt
deg/s	degrees per second
DOF	degrees of freedom
DPS	degrees per second
ECEF	earth-centered, earth-fixed
ECI	earth-centered inertial
FFT	fast Fourier transformation
FOG	fiber-optic gyroscope
FQA	factored quaternion algorithm
g	g-force
GB	gigabyte
GPS	Global Positioning System
GUI	graphical user interface
Gyro	gyroscope
HDF5	Hierarchical Data Format, version 5
hr	hour
IC	integrated circuit
IMU	inertial-measurement unit
INS	inertial-navigation system

lb	pound
MAT	MATLAB file type
microSD	micro-secure digital
MEMS	micro-electromechanical systems
N	north
NAVSTAR	Navigation System Using Timing and Ranging
NED	north–east–down
NPS	Naval Postgraduate School
oz	ounce
p	page
PNS	personal navigation system
PSD	power spectral density
rad/s	radians per second
RAM	random-access memory
RF	radio frequency
RLG	ring-laser gyro
SNR	signal-to-noise ratio
TSPI	Time-Space-Position Information
TXT	text file type
USB	universal serial bus
VIMU	virtual inertial-measurement unit
W	west
ZUPT	zero-velocity updates

## ACKNOWLEDGMENTS

I first thank my thesis advisor, Xiaoping Yun, for his expert support and consultation throughout this research. From initial ideas to review of the finished product, his thoroughness was impressive. I would be happy to be half the engineer he is.

Thank you as well to James Calusdian for his day-to-day assistance and advice. Just bouncing ideas off him improved the entire project.

I am grateful to my family for their constant support and motivation throughout my life.

My father supplied endless encouragement, gently steering me to engineering from an early age by praising my enthusiasm for problem solving: “There’s my little engineer,” he’d say—a seed planted that later blossomed.

My mother provided positive reinforcement, showing me what it looks like for a person to work harder than anyone else to accomplish personal goals.

I owe thanks to my great-grandfather, for choosing education. He was an illiterate, dirt-poor 16-year old growing up on an Alabama farm in the early 1900s, miles from the nearest neighbor. A few schoolteachers in town noticed a spark of intelligence in the dusty teenager; they tutored and encouraged him until one day he opened a letter inviting him to study at the University of Alabama. He decided to leave farm and family behind, his resolve tested by his parents’ warning, “If you walk out that door don’t bother coming back.” He never did go back, and his hard work and drive for a better life paid off. His children and children’s children prospered from his courage.

Most of all, I must single out my loving wife, Jeni, for a very special thank you. I suppose it could have been done without you, but I wouldn’t have wanted to. You make everything so much better.

THIS PAGE INTENTIONALLY LEFT BLANK

# **I. INTRODUCTION**

The work accomplished by a series of Naval Postgraduate School researchers is continued in this thesis. In addition, the enhanced position finding techniques developed for this report have already found a real-world application; they will be used as a technology demonstrator to locate dismounted infantry soldiers in a Global Positioning System (GPS) denied environment. The true number of alternate applications is only limited by imagination.

## **A. IMPORTANCE OF POSITION**

Position knowledge—or more specifically, accurate, precise, and timely position data—is essential to safe and efficient navigation in the increasingly congested air, land, and sea environments of today. Recent technological leaps in position accuracy have made guidance and navigation technologies commonplace and irreplaceable. To improve aircraft routing and de-confliction, the aerospace industry has begun replacing legacy air-traffic-control radar systems with aircraft-reported latitude, longitude, and altitude [1]. On land, smartphone-equipped drivers avoid traffic jams in real time, using downloadable software for guidance and route planning. The majority of these applications depend on the integrity of GPS signals, which the military has utilized in many vital ways—for example, to place a position-of-interest onto a military grid-reference system to call airstrikes on enemy combatants. GPS signals are physically limited and cannot penetrate many locations where position knowledge is critical to mission success.

### **1. How GPS Works**

GPS consists of L-band signals sent from satellites 20,200 km above the Earth's surface [2]. By the time these signals reach users, they are very weak, attenuated to as low as  $-158.5$  dBW [3]. A GPS receiver may be unable to acquire position data if the signal is blocked by man-made or natural obstructions such as the hull of ships, buildings, or canyons, or is lost in radio frequency interference that creates a GPS-denied environment. Four to six satellite signals are required to obtain a position fix [4]. For high

accuracy, an unobstructed path from satellite to user, with a clean signal and optimal satellite positioning, defined as a wide spread of satellites across the sky to produce a low geometric dilution of precision, is ideal [5]. In problematic environments, GPS performance can plummet from high accuracy (e.g., estimating location within 2.5 m with 95% confidence) to abysmal accuracy (results within 2,000 m). In practice, a receiver typically swings from high accuracy to no position fix, quickly skipping through the lower quality estimations. This sudden lack of service may result in confusion, reduced efficiency, and degraded situational awareness for the military user, undermining the probability of mission success.

## **2. Benefits of IMUs**

Inertial-measurement units (IMUs) are devices that contain an assortment of force feeling sensors such as gyroscopes, accelerometers, and magnetometers. The chief benefit of IMUs is their very high dependability, but they do not come near the accuracy of GPS in determining position. While GPS accuracy remains steady over time, IMU accuracy degrades because dead reckoning is used to determine location. IMU sensors that detect movement, such as accelerometers and gyroscopes, are subject to many sources of error that corrupt measurements, and errors may bleed between instruments, e.g., from gyroscope to accelerometer, causing the total errors from different sensors to multiply together and accumulate over time. Without an outside source to correct position errors, IMU performance may degrade to unusable levels. Nevertheless, IMUs have the advantage of providing position information during those times GPS cannot. A person or system using an IMU as a position reference can enter a building, ship, canyon, jammed area, or any other GPS-denied or degraded environment without fearing a sudden loss of location data.

## **3. Improving IMU Performance**

To boost the accuracy of IMUs, special processing algorithms for both real-time use and post-data collection are developed. Noisy and seemingly inaccurate data is digitally filtered to amplify desirable signals and attenuate others. Other software detects each footstep of a walking IMU user, categorizing data according to stance or swing

phase and enhancing measurements from specific sensor types (e.g., gyroscopes). Previous testing suggested that one type of sensor may perform better than others in a given gait-phase of walking. For example, a gyroscope may provide a better estimation of swing-phase measurements than an accelerometer; therefore, gyroscope data is amplified during the swing phase while the accelerometer is de-emphasized. In this way, position accuracy is significantly improved by accepting only the best data available at a given time.

Data selection is optimized by detecting the user's walking phase to reveal sensor-error biases in the stance phase. This knowledge is used to make corrections in the swing phase. In effect, what was once a debilitating negative (i.e., error bias) can now be identified and eliminated. Through this process, the user's final position, as well as his or her exact route of travel, can be estimated with surprising accuracy [6]. Three key advances in personal navigation are investigated: the use of software filtering to reduce the negative effects of sensor errors; improvement in performance owing to the optimal distribution, mounting, and selection of sensor types; and the combination of multiple co-located IMUs to enhance accuracy in a fused system. The overall purpose is to investigate the IMU as an alternative that may reduce or replace reliance on GPS.

## **B. RETICLE**

This project contributes to a larger, multifaceted effort known as "Reticle." Reticle is a network-centric warfare system developed at the Naval Postgraduate School (NPS) as a complete proof-of-concept for offloading the geometry-of-fires problem from the ground soldier to a networked system by means of a robust, lightweight, low-cost, easy-to-use solution. Reticle connects users through a network [7], determines their rifle's orientation [8], and accurately reports each user's position without the aid of GPS. The latter objective, that of freeing positional systems from GPS, is explored in this thesis.

### **1. The Geometry-of-Fires Problem**

Riflemen expend considerable effort in preventing accidental casualties by de-conflicting weapons employment among individuals within a unit and between units.

Currently, it is every soldier's responsibility to visually confirm there are no friendly forces within range of their weapons. This requires constant situational awareness and is manageable with a clear line-of-sight; however, the risk of friendly fire increases when a group of soldiers enter a disorientating environment, e.g., inside a building or ship, where bullets may penetrate a thin wall, striking friendlies. Ensuring proper geometry-of-fires is essential to completing an infantry mission safely but is complicated in its execution [8].

## **2. Reticle Subprojects**

Reticle also contains other subprojects that have been completed or are near completion by NPS students such as a battlefield-communications solution that automatically notifies a teammate or commander when a soldier fires and provides the weapon's orientation to indicate the direction of the enemy. Historically, soldiers have used radio to notify the commander when hostile forces are encountered. Delays in notification, which may postpone the arrival of reinforcements and redirect focus from the firefight, endanger troops and reduce the chances of mission success [9]. Another Reticle project improves position-finding techniques by adding special logic to determine the stance of a soldier. This posture-detection algorithm supplements existing algorithms for walking and standing by detecting kneeling and prone positions as well. Posture detection increases the suitability of IMU-based navigation by eliminating system errors in a much wider range of posture modes [10]. If a soldier lies prone for hours, for example, currently available inertial sensors steadily drift in error by several thousand meters, and the reported position diverges from the true location, rendering the system useless. With posture-detection logic, this drifting is prevented and the reported position remains steady, allowing soldiers and commanders to be confident of its accuracy.

Although this project is primarily focused on improving personal navigation, projects like Reticle exemplify what can be accomplished by achieving this objective. An accurate inertial position finding technology has the capability to increase the combat effectiveness of ground soldiers by answering vital needs identified by warfighters. With the continuing rapid advances in the performance of low-cost inertial sensors and continual miniaturization of electronic systems, it can be expected that the

geometry-of-fires problem will be solved, freeing riflemen to focus more on their mission.

### **C. PREVIOUS WORK**

In this thesis, work completed in 2010 by James Calusdian that created a personal navigation system (PNS) is built upon. In his doctoral research at NPS, Calusdian created a gait-phase detection algorithm in conjunction with custom processing functions that determine user position. These algorithms rely on measurements derived from a single IMU mounted atop the user's foot, that is, on the instep. Gyroscope measurements were relied upon heavily during the swing phase of walking, while accelerometer and magnetometer measurements were strongly weighted in the stance phase. Calusdian's PNS proved quite accurate, yielding an error of only 1.0 m after 400 m of walking. This is especially impressive considering that the IMU tested was manufactured in 2006; modern IMUs are considerably more accurate [11].

The PNS achieved a relatively high level of accuracy by significantly reducing IMU errors through an error-reduction method known as zero-velocity updates (ZUPT). With this technique, the foot-mounted IMU recalibrates every time the foot strikes the ground. The error biases identified by this software recalibration are then removed from the previous swing-phase measurement, resulting in significantly improved position accuracy.

Another set of functions was created by Calusdian to work synergistically with the ZUPT and calculate user position. The overarching algorithm to complete this task is a quaternion-based complementary filter derived from an earlier NPS-produced factored quaternion algorithm (FQA), which was modified in the course of this research to improve robustness. The FQA computes pitch and roll from accelerometer measurements and heading from magnetometers. Distance traveled is found by integrating gyroscope measurements. This strategy proved to work well [11].

## **D. THESIS OBJECTIVE**

The objectives of this thesis are fourfold:

1. To determine whether the performance of modern, low-cost IMUs has improved in the decade since the original PNS testing was completed. This is found by repeating Calusdian's experimental procedures and comparing results.
2. To determine whether distributing different sensor types by attaching them at different body locations improves performance; for example, by mounting a gyroscope on the foot and an accelerometer and magnetometer on the lower back.
3. To determine whether averaging multiple co-located sensor modules improves performance.
4. To integrate filtered position data from the sensor modules into the purpose-built Reticle software-analysis tool developed in this research.

## **II. BACKGROUND**

Key developments in the history of navigation are presented in this chapter. In the past 100 years, the field of navigation has made radical advances. Historically, reliable marine navigation for commerce and war was the primary motivation for improvement, and significant advances were few and far between. Not until the 18th century did the ingredients for a technological revolution come together.

### **A. A BRIEF HISTORY OF POSITION FINDING**

Volumes have been written about minor inventions having major impacts. From accurate timepieces for determining longitude at sea to a laser to determine orientation, each application of a new position-finding technology created a sea change within the realm of navigation.

#### **1. Longitude**

Accurate position information has been highly coveted for hundreds of years. The most concentrated effort to advance navigation occurred after four British warships sailed onto rocks near the Isles of Scilly and sank, killing 2,000. Seeking to avoid other navigational disasters, Parliament passed the Longitude Act of 1714, which offered £20,000 (\$4.8 million in today's U.S. dollars [12]) for a simple and accurate method of determining longitude.

In that era, captains at sea used a sextant to get a bearing on a celestial body and used the measurement to calculate latitude, but this solved only the north-south half of the problem. Longitude remained elusive. Captains typically had proprietary formulas for calculating their east-west positions, which resulted in a dangerous amount of overconfidence as they were usually based on constantly changing variables that might include tossing logs overboard to gauge speed, using a magnetic compass and sandglass, and interpreting currents and winds. These techniques steered thousands of sailors to a watery grave, so most oceangoing captains kept to heavily traveled routes, reaching their destination by cruising to the target latitude and then turning east or west as required. The

routes attracted pirates and were inefficient, since a constant latitude on a sphere is longer than a great-circle route [13].

## **2. The Harrison Sea Clock**

The solution to longitude came from the English clockmaker John Harrison, who presented his first sea clock to Parliament in 1730 and refined the design over 40 years. Accurate to within one second a month, Harrison's sea clocks were a tremendous improvement over contemporary timepieces, the best of which drifted 30 minutes per month. To determine longitude, the Harrison clock was synchronized with an authoritative land clock—typically at the Greenwich or Paris observatories. Once underway, the captain compared local noon (the sun's highest point) to the time on the sea clock. Since the earth rotates  $15^\circ$  every hour, he subtracted the time from 12:00 and multiplied the decimal difference by 15 to find the degree of longitudinal change. Clock-derived longitude was combined with sextant-derived latitude to determine coordinates [13].

Knowing latitude and longitude was sufficient for travel by sea and land, but with the invention of aircraft, more than just position was needed since a pilot must also know attitude.

## **3. Early Mechanical Spinning Gyroscopes**

In 1944, the final years of World War II, German scientists made use of two mechanical spinning gyroscopes to stabilize the pitch and heading of a V-2 rocket along with an accelerometer to determine velocity. Large spinning gyroscopes had been in use for several years to steady ships per the conservation of angular momentum, but the addition of an accelerometer was revolutionary because it enabled distances to be measured. Although rudimentary by today's standards, the accelerometer on the V-2 was effective in determining the speed of the rocket. Calculations were performed prior to launch to determine an engine cutoff speed that would enable the rocket to glide directly to the target, and the result was programmed into the V-2's onboard analog computer. This surprisingly accurate missile struck within 6 km of a programmed target 50% of the time when launched from a distance of a few hundred kilometers [14].

Another German application of spinning gyroscopes was the V-1 flying “buzz” bomb, which used a rudimentary autopilot consisting of two mechanical gyroscopes spun up with compressed air to control pitch and yaw. An anemometer on the nose determined whether the bomb had traveled the set distance; if so, it dove steeply, typically landing within 11 km of the target. This level of accuracy is rough but was capable of terrorizing civilians in a metropolis such as London. Guidance systems from unexploded V-1s were recovered, reverse engineered, and used as a template for the American LTV-N-2 Loon rocket. A Loon guidance system is housed at NPS, as seen in Figure 1.

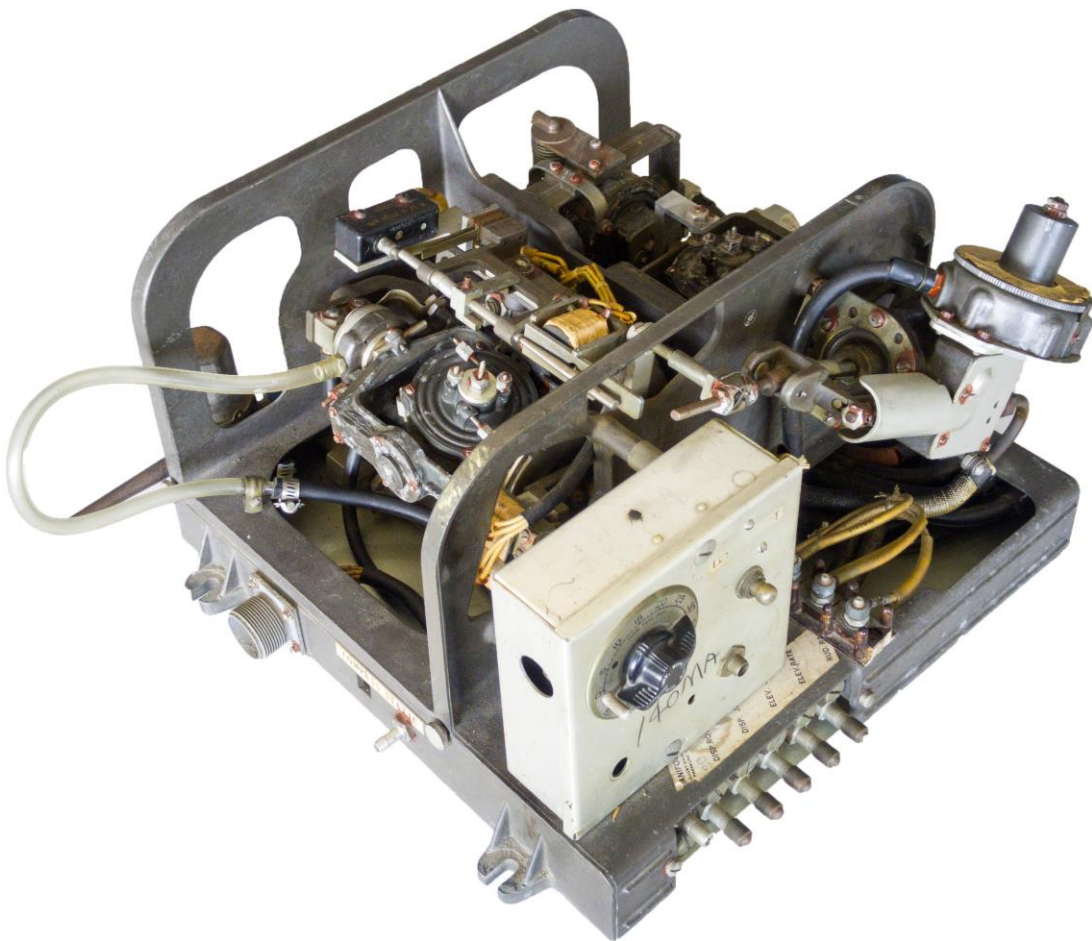


Figure 1. LTV-N-2 Loon Rocket Guidance System

After the war, Operation Paperclip brought German guidance technology to the U.S. by importing scientists such as Wernher von Braun to integrate German innovations into the U.S. arsenal [15]. Mechanical gyroscopes eventually became quite reliable and accurate, and their application transitioned from stabilizing airborne platforms to being packaged into an IMU with accelerometers as a navigation aid. These IMUs, commonly called inertial-navigation systems (INS), provided true heading and position estimates more accurately than ever before. Despite delivering such a useful capability, mechanical IMUs had several drawbacks. The spinning gyroscopes were large and heavy, typically around 23 kg (50 lb) and required alignment times around 30 minutes—undesirable for use in fighter aircraft, which have limited space, fly best when light, and must scramble start and launch on short notice. A shorter alignment can be accomplished with a stored-heading alignment but lacks the performance seen in the full performance gyrocompass alignment. Any replacement for the mechanical gyroscope had to be lighter, equally reliable, require short alignment times, and match or exceed accuracy. The solution was the laser-light gyroscope.

#### **4. Laser-Light Gyroscopes**

Modern inertial-navigation systems were developed in the 1980s with aerospace applications of the ring-laser gyroscope (RLG). This non-spinning, gimbal-less gyroscope can be implemented as a “strap-down” system [16]; unlike a spinning gyroscope of fixed orientation with respect to earth, a laser-light gyroscope can be bolted to an aircraft’s airframe and move with it. The gyroscope no longer needed to be fixed to an earth-centered inertial (ECI) reference but is now oriented to the variable movement of the aircraft itself as it maneuvers. The INS converts measurements to a local fixed reference frame such as earth-centered, earth-fixed (ECEF) through onboard computers. The difference between gimbaled gyroscopes and RLGs is not limited to initial frames of reference, and their technologies for detecting rotation are vastly different [17].

RLGs operate by determining a frequency difference between counter-propagating laser-light beams, splitting the laser beam and sending the split light in opposite directions around a triangular or square closed path (or “ring”) of

low-expansion glass. Mirrors at each corner redirect the laser to another mirror until the light reaches the starting point and light waves are allowed to interfere with each other in front of a sensor. When the RLG is stationary, both split beams have the exact same frequency. As it rotates, the frequency of the laser light increases or decreases depending on the direction of rotation. If the direction of the gyroscope's physical rotation is the same as the laser's, the light frequency increases, if opposite, the frequency decreases. This causes a beat frequency, a slight frequency difference that causes destructive interference and a fringe pattern of light. The beat frequency is proportional to the rate of rotation, and the direction of the interference fringe pattern correlates to the direction of rotation. This process is precise. As a strap-down system, RLGs can directly measure own-ship angular rates. Modern medium-grade RLGs tend to drift 0.5–1.0 nautical miles per hour, weigh 10–50 pounds, and usually cost more than \$50,000. The widespread adoption of RLGs is currently limited by their high cost, which reflects stringent manufacturing specifications and the need for a high voltage power source to produce the laser [18], [19].

A slightly less expensive and newer gyroscope called the fiber-optic gyroscope (FOG) also uses laser light to determine angular rate. Like the RLG, the FOG measures the phase shift of a counter-rotating laser beam but contains up to 5,000 m of optical fiber, in contrast to an RLG's path length of centimeters, theoretically enabling better measurement resolution. The FOG's development occurred after the RLG, and its technology was not sufficiently reliable for INS applications until the late 1990s. They are still maturing, with smaller size and cheaper manufacturing costs incentivizing their development [20].

Laser-light gyroscopes may be coupled with accelerometers to produce a dead-reckoning solution for navigation, but over a few hours, even the best INS tends to drift by a few kilometers. This increase of position error over time is a persistent problem with INS technology. When first used in aircraft, before GPS was available, special procedures were followed to offset cumulative error, but as these fix-taking methods depended on variables such as pilot technique and low altitude flyover of a reference point, they were not accurate. Because this method only updated position and did not

reduce drift-rate errors, multiple fix-taking attempts were required for each flight depending on the mission. Soon after the adoption of RLGs, another development emerged to fill INS performance gaps. This new space-based system revolutionized navigation by providing extremely accurate position updates.

## **5. GPS**

The most visible 20th-century improvement in navigation technology is GPS, based on the “navigation system using timing and ranging” (NAVSTAR) program of the late 1970s. The system was made available for public use through a 1983 U.S. presidential directive. By the early 1990s, GPS satellites in orbit provided worldwide coverage, and in 2000, a feature known as selective availability, which purposely degrades the unencrypted L1 civilian signal to a position error of about 100 m, was turned off, again by presidential directive [21].

GPS position measurements are more accurate than INS and do not drift over time. Federal Aviation Administration studies in 2014 showed a Rayleigh distributed horizontal error of less than 3.351 m 95% of the time, as averaged from 28 sites worldwide. Most measurements from this report showed errors of less than 2.0 m, indicating the user could generally expect even better performance [22].

The robustness and accuracy of the GPS constellation is continuously improved as new satellites with better technology replace legacy platforms that have exhausted their life cycle. Upgrades include broadcast capability for additional unencrypted GPS signals to supplement the existing 1575 MHz L1 signal. These new signals—including the 1227 MHz L2C, 1176 MHz L5, and an improved version of L1 called L1C—offer greater redundancy and accuracy, which is achieved by transmitting multiple signal frequencies at higher effective powers with improved signal structures. Future civilian receivers are expected to achieve sub-meter accuracy when the current civilian L1 signal is processed with at least two other signals (e.g., L2C and L5) through a technique called trilateration [23]. If the past is any indication of the future, the entire world will welcome the benefit of improved GPS position accuracy and reliability.

Despite efforts to improve GPS through increased effective signal power, modernized code structures, and additional frequencies, GPS remains limited by the physics of radio waves. Obtaining a reliable position depends on the device's ability to receive unobstructed line-of-sight signals from at least four satellites orbiting at an altitude of 20,200 km in space, and there will always be countless scenarios in which a GPS receiver is blocked and rendered useless. Receivers require that GPS satellite signals have a specific carrier-to-noise ratio ( $C/N_0$ ), or equivalently a signal-to-noise ratio (SNR), in order for the receiver to obtain the position. Severe signal attenuation can occur under roofs, in urban canyons beside large buildings, under dense foliage, under water, or in tunnels. Alternatively, a GPS signal may not be attenuated, but the noise floor may be raised through intentional or unintentional GPS jamming; such radio-frequency interference can degrade the SNR such that the GPS signal is completely masked by artificial noise. Due to these problems in signal continuity, GPS cannot be relied upon as the sole navigational reference for a military personal navigation system (PNS).

Although GPS is low-cost, light, and accurate, the fact that it can so easily be disrupted requires an additional positioning source. This problem is partly solved by synergistically blending GPS signals with INS measurements using a Kalman filter. Many air- and land-navigation systems incorporate a proprietary, blended-solution Kalman filter. Unfortunately, a laser-light INS is not practical in a man-portable navigational system because it is cumbersome, heavy, and expensive where a tiny, passive, low-cost solution is needed.

## **6. MEMS**

The invention of micro-electromechanical systems (MEMS) has enabled the size of IMUs to be significantly reduced. MEMS miniaturization is accomplished by manufacturing the electronic and mechanical components on the same wafer. Like their larger cousins, MEMS measure physical phenomena but in a much smaller package, usually ranging from 1.0 mm to 0.001 mm. A key benefit of MEMS technology is that components can be manufactured very inexpensively [24]. One of the first implementations of MEMS technology was made in 1991, when an accelerometer was

integrated into a crash-detection system to sense vehicle collisions and trigger airbag detonation [25]. Other sensor types added to MEMS chips include the triaxial gyroscope—where three sensors of the same type are mounted orthogonally, allowing measurements along all three axes (e.g.,  $X$ ,  $Y$ , and  $Z$ ). A triaxial gyroscope combined with a triaxial accelerometer allows six-degrees-of-freedom (6-DOF) measurements, specifically of linear acceleration and angular velocity. In this research, we utilize a nine-degrees-of-freedom (9-DOF) IMU, which adds a triaxial magnetometer to measure magnetic fields. Modern manufacturing methods have led to MEMS IMUs, which are much smaller and cheaper than RLG- and FOG-based systems but with comparatively poor accuracy—at least an order of magnitude inferior to non-MEMS counterparts.

A chief benefit of MEMS-based IMUs is low cost. Prices plummeted with the worldwide adoption of smartphones, beginning with the Apple iPhone of 2007, which included a MEMS accelerometer. As consumers demanded more, manufacturing ramped up, production was streamlined, yield rates increased, and unit cost fell. In 2010, Apple added a MEMS gyroscope to the iPhone 4 to improve gaming performance, and most smartphones now have at least a 6-DOF sensing capability [26]. At the present time, a MEMS-based 6-DOF IMU can be purchased for under ten dollars [27]. Unfortunately, IMU performance has not increased as dramatically as prices have decreased, and PNS applications require performance-enhancing filtering techniques to deliver adequate accuracy.

## **B. IMU CHARACTERISTICS**

IMUs are sensors that detect linear acceleration, rotational velocity (i.e., turning), and (depending on the model) magnetic orientation. Several companies combine MEMS-based accelerometers, gyroscopes, and magnetometers onto one integrated circuit (IC) board, currently available to consumers for \$10 to \$3,000 [27], [28]. There is a moderate correlation between the price of a sensor module and its advertised performance specifications. A few companies bundle software with their IMUs to provide a graphical user interface (GUI) for settings, calibration, and data-recording. Some products include onboard (i.e., Kalman) filtering for the IMU to create an attitude heading and reference

system (AHRS) [29]. One company offers a proprietary datalink to connect and synchronize several distributed sensors and telemeter their measurements to a waiting application [30]. As with most technologies, the price of raw sensor components has dropped over time as performance has improved.

MEMS IMUs are essentially a miniaturized version of conventional sensor units. Most MEMS accelerometers are piezoelectric, acting as a transducer by converting a physical force exerted on a mass into an electric signal. The signal is produced by the piezoelectric material that suspends the mass, which proportionally reacts to the experienced linear acceleration [31]. Many MEMS gyroscopes convert angular velocity into an electrical signal through a similar transducer concept. Rotation causes physical displacement of a vibrating structure such as a tuning fork, creating an electrical signal correlated to a rate of rotation. Departing conceptually from these technologies, MEMS magnetometers determine a magnetic orientation by measuring changes in resistivity in a thin ferrous element whose resistance changes with the magnitude of a magnetic field perpendicular to the direction of current [32].

These sensors applications represent only a fraction of the MEMS technologies now found in the marketplace. As new methods and improvements evolve, IMU price reduction is expected to continue, accompanied by a steady rise in performance. Companies that create IMUs and AHRSs periodically survey the status of MEMS sensor technology. When improvements reach the price threshold for a targeted market, they create a new IMU with updated MEMS hardware and filtering software to increase performance and reduce cost. Overall, the consumer benefits greatly from intense market competition.

### **C. IMU/ SENSORS USED IN THIS RESEARCH**

MEMS-based sensor modules were acquired from YOST Labs (formerly part of YEI Technology). Sold at \$255, the YEI 3-Space Data-Logging v2.0 AHRS is a sensor-fused module combining a triaxial accelerometer, gyroscope, and magnetometer onto one integrated circuit (IC). As shown in Figure 2, the module is about the size of

two universal-serial-bus (USB) flash drives placed side by side, measuring 35 mm × 60 mm × 15mm (1.4 in × 2.4 in × 0.6 in) and weighing 28 grams (0.98 oz).



Figure 2. YEI 3-Space Data-Logging v2.0 AHRS

The sensor module uses triaxial sensors and detects acceleration (via accelerometer), rotational velocity (via gyroscope), and magnetic orientation (via magnetometer) with respect to three dimensions. The module also provides a user option to send raw sensor data through a YEI proprietary onboard Kalman filter, resulting in smoother, more accurate outputs. The 3-Space module does not require a connected power source during operation, owing to a built-in rechargeable lithium battery with three hours of useful life. For all experiments, data were recorded in the included two gigabyte (GB) micro-secure-digital (microSD) card. Data from each test was recorded in a text file (TXT) and transferred to computer through a supplied micro-USB to USB cable. Additional tests were recorded to a file without overwriting previous recordings. The data-logging 3-Space module paired well with the investigations conducted for this

thesis. The only substantial problem with this sensor module was the laborious post-processing required to synchronize data from multiple simultaneously recorded sensors. A wireless version of the 3-Space module was available but lacked onboard data logging and simultaneous reception from multiple sensor modules; therefore, the data-logging variety was used exclusively.

This research did not seek to create a Kalman filter for an IMU, as this was available through YEI technology. Apart from sensor characterization tests, raw sensor data were not recorded, and proprietary YEI Kalman filter output was saved onto the microSD card.

The accelerometer integrated by YEI technology into the 3-Space module was the MMA8451Q digital accelerometer manufactured by NXP semiconductor. This triaxial MEMS accelerometer senses linear accelerations of  $\pm 2$  g,  $\pm 4$  g, or  $\pm 8$  g, depending on the selected setting. The  $\pm 2$  g setting provides 4096 counts/g, and the  $\pm 4$  g setting, 2048 counts/g [33]. This indicates higher precision in the measurement. It was originally assumed that the highest precision outputs offered by the  $\pm 2$  g setting would give superior performance, but this proved false. Problems with sensor saturation emerged when the force encountered went above 4 g; therefore, the  $\pm 8$  g setting with 1024 counts/g was used, resulting in a resolution of roughly 0.00096 g or  $\sim 1\text{cm/s}^2$ . Though this setting does not have the highest resolution, it offers more than adequate performance.

The unit “g” represents g-force, which is defined as  $9.8065\text{ m/s}^2$  and does not change with location [34]. In Monterey, California the local g-force is equal to  $9.7991\text{ m/s}^2$  [35]. Therefore, the local g-force is:  $9.7991/9.80665 = 0.99923\text{g}$ .

The MEMS gyroscope used in the 3-Space module was the Maxim Integrated MAX21000, described by the manufacturer as an ultra-accurate, low-power, 3-axis, digital-output gyroscope [36]. Like the accelerometer, the gyroscope featured a triaxial configuration, allowing the measurement of rotational velocity around three axes. The gyroscope had several degrees-per-second (DPS) settings to choose from in the 3-Space configuration file, ranging from 250 DPS to 2,000 DPS. Note that the abbreviation “DPS” is used extensively for gyroscope specifications in place of the abbreviation “ $^\circ/\text{s}$ ”

or “deg/s.” In this report, those three abbreviations all represent a unit of degrees per second. Testing revealed that any setting less than 2,000 DPS resulted in inaccurate data, owing to sensor saturation, which occurred when a foot-mounted sensor module was walked down a straight, long hallway. Saturation was easily identified in gyroscope magnitude data by a plateauing effect in the measurements, in which the maximum value was restricted to the corresponding gyroscope setting (e.g., 250 DPS). If only the step-plot was monitored, saturation was not nearly as recognizable because the overall shape of the test run remained the same. The distance between each footstep that exceeded the DPS setting was reduced because of the angular-rate cutoff, which had the effect of reducing total distance traveled without affecting the direction of travel. When each step was integrated, the resulting distance estimate was incorrect, though the shape of the plot matched the user’s traveled route. For example, if the straight, long fifth-floor hallway of the engineering building was walked with a sensor module on the foot, the processing algorithms estimated the path length at 135.0 m instead of the true 228.0 m. The plot was straight, but the scale was reduced. These saturation problems were prevented by selecting the gyroscope’s highest measurement setting of 2,000 DPS, which yielded a resolution of 15 digits/DPS as given in the supplied YEI technology data sheet. This is equivalent to a resolution of  $0.0\bar{6}$  DPS about each axis [36]. Gyroscope performance is more important than accelerometer or magnetometer for personal navigation, and this phenomenon is explained in the next section.

The MEMS magnetometer used in the 3-Space module was the Honeywell temperature-compensated HMC5983 three-axis digital compass IC [37]; in this thesis, “magnetometer” is synonymous with “compass”. The magnetometer was designed for use in a low-field magnetic environment, such as that typically seen in personal navigation applications. YEI lists the 3-Space module as having a  $1^{\circ}$  to  $2^{\circ}$  heading accuracy with its HMC118X magnetoresistive sensors. As with the accelerometer, lower maximum sensor settings result in higher-precision readings. To find a balance between resolution and range, multiple tests were completed to find the lowest magnetometer setting that could be used without causing sensor saturation. These tests were performed in a long fifth floor corridor inside the engineering building, because it was assumed that

this environment, including electrical wires passing current within the walls, would provide a suitable peak magnetic value. The lowest available setting, 1.9 Gauss, was determined suitable, as it did not saturate the magnetometer. The resulting digital resolution at that setting was 1.22 mili-Gauss per least significant bit over a range of  $-2048$  to  $2047$  bits [37]. The integration of a magnetometer for the PNS is regarded as a stopgap solution to provide heading measurements until such time as MEMS gyroscope heading accuracy is improved, as discussed in detail later.

#### **D. GYROSCOPE ERROR SOURCES**

IMUs are subject to errors from several sources. For the accelerometer and gyroscope, these errors may initially appear small but accumulate over time, growing somewhat linearly into large position and heading errors. Better performance comes at a price, however, as a medium-grade INS can cost almost \$100,000. Compromises were made for each sensor type's price versus performance versus size and weight. A \$70,000 INS that weighs 40 lb (18 kg) may be well suited for a B-52 bomber but unrealistic for each Marine in a fire team to carry on his or her back; thus, it makes sense to direct limited funds toward sensors that provide a good ratio of cost to performance and suitability. To maximize cost versus benefit, research was conducted to identify the largest source of errors in MEMS IMUs; Berman found the dominant source of IMU errors to originate from the gyroscope [38].

Gyroscope errors can be classified by type, with each type capable of degrading accuracy more than any other kind of IMU sensor, e.g., the accelerometer. Since a complementary filter relies mostly on the gyroscope to calculate position changes during the swing phase, it is important to identify and take steps to mitigate these major error types [11]. In descending order of prevalence, they are error bias stability, scale-factor error, including non-linearity and asymmetry, and angular random walk (ARW)

As a whole, those three sources of gyroscope error were found to be larger contributors to IMU degradation than any type of accelerometer error. Unfortunately, a magnetometer was not evaluated in his study. Unlike gyroscopes, a magnetometer's heading error remains stable with time. Gyroscope error bias stability, where most of the

error occurs within the first 100 seconds of operation, was found to produce an angular error almost five times larger in magnitude than scale factor error and nearly 14 times larger than ARW [38]. The next few paragraphs define these error types.

## 1. Error Bias Stability

Error bias stability is defined as a fluctuating amount of error in relation to the true value over time. Instead of holding a constant bias, flicker noise causes the error bias to randomly wander within a bounded area. For example, if the gyroscope was stationary and measurements were taken, it should yield a measurement of 0 deg/s. Since there is a fluctuating error bias, it may read 0.2 deg/s a moment after power on, increase to 0.4 deg/s a few seconds later, and then slowly taper off to 0.1 deg/s as time exceeds 100 s. The velocity measurements are integrated once to output an angle in degrees, so the fluctuating error bias causes an unsteady increase in the amount of angular error built up over time [39].

Flicker noise is the cause of error bias stability issues and is dominant at low frequencies where the power of its degrading effects can be modeled as  $1/f$ , where  $f$  is frequency. Error bias stability is different than ARW because the dominant error source for the former is flicker noise at low frequencies, and the dominant error source for the latter is high frequency white noise. Error bias stability can be modeled as a bias random walk (BRW) for times of less than 100 seconds,

$$\text{BRW}(\text{°} / \sqrt{\text{hr}}) = \frac{\text{BS}(\text{°} / \text{hr})}{\sqrt{t(\text{hr})}} \quad (2.1)$$

where BS is defined as bias stability, and time  $t$  is usually listed in hours and can be found in the manufacturers specifications [39]. Units of seconds may be more appropriate considering this model's applicability is limited to about 100 s, but manufacturer specifications use hours, so it was decided to use the same units throughout.

## 2. Scale-Factor Error (Non-Linearity and Asymmetry)

Scale-factor error is an error that arises when the measurement output is not proportional to the input force. For example, if a gyroscope rotated at a linearly

increasing velocity (i.e., acceleration), then the slope of the measurement outputs do not match the slope of the input force. Scale-factor can have both a linear and non-linear component as well as an asymmetry such that the shape and slope of the measurement is different depending on the direction of rotation. These assortment of errors are grouped into the term scale-factor error, which occur due to manufacturing tolerances, aging of the sensors, and imperfections with the signal amplifiers [40].

### 3. Angular Random Walk

Angular random walk is a gyroscope angular rate error caused by thermo-mechanically created high-frequency white noise [41]. When discussing inertial sensors, another word for noise is “drift.” In other words, the gyroscope measurements appear to drift from the true value. The purpose of an ARW specification is to use them to find the standard deviation, or spread, of noise-induced random error as it grows proportionally with the square root of time. Some manufacturers list this error as a noise density in units of  $^{\circ}/\text{s}/\sqrt{\text{Hz}}$ , which can be converted from power spectral density (PSD) to ARW with [41]

$$\text{ARW}(^{\circ}/\sqrt{\text{hr}}) = \frac{1}{60} \sqrt{\text{PSD} \left[ \left( \frac{^{\circ}}{\text{hr}} \right)^2 / \text{Hz} \right]}. \quad (2.2)$$

For the KVH 1775 FOG gyroscope, whose ARW is listed as  $0.012^{\circ}/\sqrt{\text{hr}}$ , after two hours, the one sigma standard deviation of the orientation error is  $\sqrt{2\text{hr}} \times 0.012^{\circ} = 0.017^{\circ}$  [39]. Compare this ARW specification and resultant orientation error to the 3-Space sensor’s less precise ARW specification of  $0.45^{\circ}/\sqrt{\text{hr}}$  and a subsequently poorer one-sigma standard deviation orientation error of  $\sqrt{2\text{hr}} \times 0.45^{\circ} = 0.636^{\circ}$ . Working with higher-quality sensors, where higher quality is defined as having a low noise level (i.e., ARW), we are more likely to observe consistent and better gyroscope performance. Better performance from the sensor enables the creation of a more accurate orientation solution.

#### 4. Other Gyroscope Errors

Gyroscopes are also susceptible to other types of error, including those induced by a faulty calibration process. Calibration error usually results in imperfect angular-rate estimates, whose error magnitude is proportional to the true angular velocity. Scale-factor non-linearity error may accompany calibration error. As a realistic example, a gyroscope might turn with an angular rate of  $100^\circ/\text{s}$  but incorrectly report it as  $102^\circ/\text{s}$  or turn at  $200^\circ/\text{s}$  but, owing to bias error, report  $204^\circ/\text{s}$ . Integrated over time, calibration error results in degraded orientation accuracy. Since currently available MEMS gyroscopes do not offer good accuracy, performing the best possible calibration is essential in creating a usable device for personal navigation [39].

The 3-Space sensors used in this research were calibrated using the YEI 3-Space Sensor Suite v3.0r7. A screenshot of the software interface is presented in Figure 3.

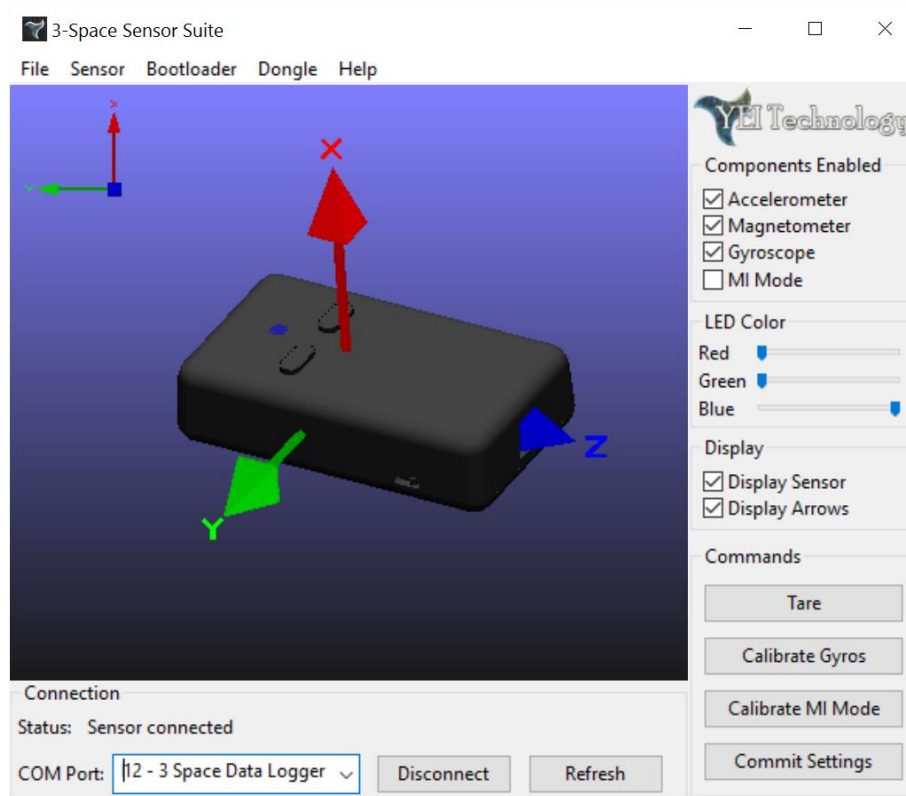


Figure 3. YEI 3-Space Sensor Software Suite v3.0r7

Note the direction of the axes protruding from the 3-Space sensor as the X, Y, and Z-axis convention with respect to the sensor module was consistent through all testing. Calibration is performed by connecting a sensor module to the computer via the USB port, placing the module on a stationary surface, and clicking “Calibrate Gyros.” Calibration is near instantaneous, a vast improvement on navigation-grade laser-light gyroscopes, which typically take four minutes or longer to align.

Temperature issues can also cause gyroscope errors. Manufacturers typically list gyroscope temperature sensitivity in units of percent per degree Celsius (%/°C). This type of scaling factor changes with temperature. MEMS sensors exhibit a non-linear relationship between sensor error bias and temperature. To counter this, most manufacturers use previously collected temperature versus error data along with an onboard temperature sensor to cancel out temperature errors [39]. The 3-Space sensor module employed had an internal temperature-correction capability to compensate the data as temperatures changed [42]. The author attempted to control the negative effects of temperature by performing most testing after sunset on an athletic track, resulting in a temperature range of 13° to 20°C (55° to 68°F) during testing. Since the 3-Space module’s black plastic housing never noticeably heated when operating, no effort was made to examine the temperature effects of current draw; the internal sensor temperature was assumed to remain stable and close to ambient. The software-run temperature corrections and the operator’s ability to calibrate sensors at or near ambient temperature were expected to reduce any overall effects related to temperature. While sensors for personal navigation should work wherever humans can go, whether desert or arctic tundra, for a proof of concept, controlling conditions is considered acceptable.

## **5. Gyroscopes in IMUs**

The main benefits of a MEMS gyroscope IMU over a RLG or FOG is they are less expensive, smaller, lighter, and consume less energy. The drawback is that MEMS IMUs are significantly less accurate. Specifications and error rates for gyroscopes across a range of currently available IMU sensor modules are shown in Table 1.

Table 1. Gyroscope Performance of Competing IMUs

	<b>KVH 1775 (FOG gyro) [43]</b>	<b>MicroStrain 3DM-GX1 (MEMS gyro) [44]</b>	<b>MicroStrain 3DM-GX4-25 (MEMS gyro) [28]</b>	<b>XSENS MTw (MEMS gyro) [45]</b>	<b>YEI 3-Space Sensor v2.0 (MEMS gyro) [36]</b>
<b>IMU Size (mm)</b>	88.9 diameter $\times$ 73.7 height	$65 \times 90 \times 25$	$36.0 \times 24.4 \times$ 11.1	$34.5 \times 57.8$ $\times 14.5$	$35 \times 60 \times 15$
<b>IMU Weight</b>	700 g (1.54 lb or 24.7 oz)	75 g (2.6 oz)	16.5 g (0.58 oz)	27 g (0.95 oz)	28 g (0.98 oz)
<b>Bias Stability</b>	0.05°/hr	70°/hr	10°/hr	20°/hr	4°/hr
<b>Scale Factor Non-Linearity</b>	0.005%	0.7%	0.02%	0.1%	0.2%
<b>Angular Random Walk</b>	0.012°/√hr	3.5°/√hr	0.3°/√hr	3°/√hr	0.45°/√hr
<b>Year Introduced</b>	2014	2006	2014	2012	2014
<b>Cost</b>	< \$20,000	~ \$3,000	~ \$3,000	~ \$1,000	~ \$250

For two of the listed IMUs, angular random walk is derived from a PSD specification of  $0.005^\circ/\text{s}/\sqrt{\text{Hz}}$  provided by MicroStrain 3DM-GX4-25 [28] and a PSD specification of  $0.05^\circ/\text{s}/\sqrt{\text{Hz}}$  provided by XSENS MTw [45]. Conversion from PSD to ARW is accomplished with Equation 2.2 [41].

The first IMU examined in Table 1, the KVH 1775 IMU, offers substantially better gyroscope performance than rival units but at a considerably higher price, reflecting the presence of a miniaturized FOG integrated into an IMU with MEMS accelerometers and magnetometers. Due to its price and bulk, the KVH 1775 is not well suited to man-portable applications but is included here as an example of a potential high-end performance solution; although, it might be more fairly compared with other FOG-based IMUs, not MEMS.

Next, the MicroStrain 3DM-GX1 is the IMU used in [11] and provides an early example (2006) of a commercial MEMS IMU assembled from individual sensors. Before its availability, researchers had to spend considerable effort in assembling custom IMUs from raw components. The 3DM-GX1's fourth-generation successor, the 3DM-GX4-25, offers far superior performance for the same cost.

The XSENS MTw IMU wirelessly streams synchronized measurement data to a computer for recording, allowing multiple sensor modules to be fused together for enhanced post processing regardless of their mounted location on the body.

The YEI 3-Space Data-Logging v2.0 AHRS sensor module was selected for use in this research as offering the best tradeoff of cost and performance. The ability to record data onto a microSD card made it possible to conduct testing without any wiring obstructions, and the performance specifications were quite good for the cost, offering better specifications in some areas than more expensive IMUs.

For this research, David Arch, marketing and project manager for Honeywell Aerospace, suggested that a gyroscope error bias stability of less than  $0.1^{\circ}/\text{hr}$  might be required to track a soldier in the field [46]. The problem of gyroscope bias stability is illustrated in a simple scenario: if an IMU is motionless for a long time, with noise removed, its gyroscope error is less than  $0.1^{\circ}/\text{hr}$ . So if a stationary gyroscope initially had a heading of  $30.0^{\circ}$ , and assuming no other errors were present, after an hour the heading could be  $30.1^{\circ}$  [47]. The FOG of the KVH 1775 IMU meets the gyroscope error bias stability threshold with an error of only  $0.05^{\circ}/\text{hr}$ , a level no currently available MEMS IMU can achieve [43]; but, as noted, the KVH 1775 is not man-portable. In the future, a low-cost MEMS gyroscope will likely break the  $0.1^{\circ}/\text{hr}$  threshold. Until then, other sensor types such as magnetometers can be used as a stop-gap solution for heading measurements. Unfortunately, position estimates must still rely on the gyroscopes, and special measurement-filtering techniques are required to make personal navigation possible.

## E. HIGH-ACCURACY POSITION USING LOW-ACCURACY SENSORS

In developing a practical PNS from MEMS IMUs, the optimal mounting location for the gyroscope, accelerometer, and magnetometer is a chief concern. Special processing software written for [11] was adapted in this research to work with the YEI 3-Space sensor module, and additional software was created to allow the incorporation of multiple sensors distributed throughout the body (e.g., at the shin or chest) as well as redundant mountings at the same location.

### 1. MEMS IMU Size Comparison

The contrast between the \$255 YEI 3-Space Data-Logging v2.0 sensor module used for experimentation and the \$3,000 MicroStrain 3DM-GX1 used by [11] nearly a decade prior is a good example of the evolution of performance, miniaturization, and affordability over time. A size comparison of these sensor modules and the fourth-generation MicroStrain 3DM-GX4-25 sensor module is shown in Figure 4. The 3DM-GX4-25 is smaller than the YEI product due to lack of an onboard battery and memory card and is also significantly more expensive.



Figure 4. IMU Form Comparison From Left to Right: YEI 3-Space Module, 3DM-GX1 (first generation), and 3DM-GX4-25 (fourth generation)

## **2. Personal Navigation System**

In the past 15 years, several reports have been published concerning the application of MEMS IMUs in man-portable navigation systems. Almost every investigator presents original techniques to accomplish this task. All used a combination of gyroscopes, accelerometers, and, to a lesser degree, magnetometers. The main focus of these reports is filtering; nearly all investigators use some form of an optimal estimator (i.e., a Kalman filter) to achieve best performance.

In this thesis, we depart from the Kalman-filter solution, building on a filter developed by NPS researchers over the past decade and continuing the research of [11], where a PNS that is executed with MATLAB code during post processing, is described as

Accelerations induced by natural walking motion will be processed to derive an updated position of the user. The strapdown navigation algorithm will be adapted for this application. It will utilize an adaptive-gain quaternion-based complementary filter that was specifically tailored for the PNS. Furthermore, the strapdown algorithm incorporated the concept of zero-velocity updates and a custom gait-phase detection algorithm to determine the instances of the foot swing and stance periods. [11]

Some of Calusdian's experiments were repeated with the modern YEI 3-Space IMU, with custom tuning of the PNS software required to extract the best possible measurements. In this research, we explore the performance advantages of fusing multiple sensors and processing the averaged measurements through the PNS, taking the code created for [11] and modifying it to allow integration into the Reticle analysis GUI created by the author. The Reticle GUI's home screen is shown in Figure 5.

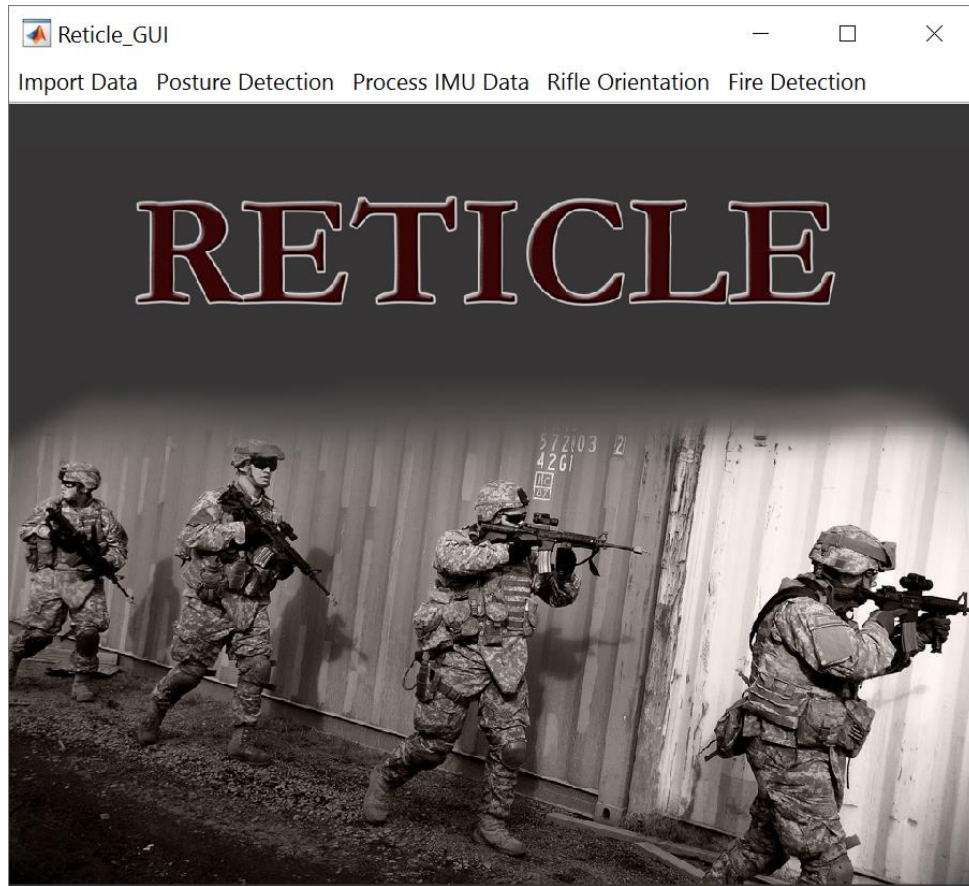


Figure 5. Reticle GUI Home Screen. Adapted from [48].

The Reticle GUI simplifies and streamlines the processes required to import data, run the PNS algorithm, and invoke functions as needed. It also incorporates other Reticle projects into the PNS such as the posture-detection algorithm created by [10].

### 3. Zero-Velocity Updates

At the heart of the PNS are gait-phase detection and zero-velocity updates. The human gait cycle consists of the swinging motion made by a person's legs between footsteps and subsequent footfalls. The swing phase begins when the IMU-mounted foot leaves the ground and swings in front of the walker and ends when the heel of the same foot strikes the ground. The forces within the swing phase, such as the impulse of a heel strike, are sensed by an IMU strapped to the top of the foot or mounted on the shin, waist, or chest.

The stance phase is defined as occurring when the IMU-bearing foot is planted on the ground, which occurs between footsteps and when standing. This phase, which is assumed to have virtually no velocity, accounts for around 60% of the walking gait cycle. Without question, movement occurs as the heel rolls to the ball of the foot or the person pivots on heel or ball to change direction, but the magnitudes are small enough to be assumed zero and still allow the ZUPT to be effective. In this thesis, the “near-zero” stance-phase descriptor is simplified to zero, with the qualification implied. As defined for this research, the job of the ZUPT algorithm is to identify velocity-error biases in the gyroscopes and eliminate them from previous swing-phase measurements. To accurately identify error biases in the field, a Time-Space-Position Information (TSPI) truth source is required to enable comparison between an IMU’s error-laden measurements and those of the actual velocities experienced. Since it is unrealistic to arrange such a setup, a simpler solution is needed. This solution weighs heavily on the assumption that the only part of a person’s gait cycle in which velocity can be known for sure is the stance phase, where foot velocity is assumed zero. This belief is essential, because the grounded, stationary foot acts a stance-phase pseudo truth source.

MEMS sensors have bias and drift errors that cause them to falsely report non-zero rates though the IMU is stationary. The ZUPT algorithm used in the PNS uses this falsely reported gyroscope data to identify and eliminate navigation-frame IMU velocity errors in the gyroscopes that were persistent during the previous swing phase. Each iteration of the PNS algorithm begins with navigation-frame acceleration measurements. These measurements are represented through

$$a_m(t) = a_a(t) + \varepsilon, \quad t = [0, T] \quad (2.3)$$

which illustrates how the true acceleration  $a_a(t)$  can be accompanied by an error bias  $\varepsilon$ , resulting in a falsely reported acceleration  $a_m(t)$ . If no adjustments are made and the reported acceleration  $a_m(t)$  is integrated over a person’s total gait cycle  $t = [0, T]$ , producing a calculated velocity  $v_c(t)$ , then the resulting position error grows, making it appear that the individual is moving at a different rate than reality. To eliminate error and

find the true velocity  $v_a(t)$ , the error bias term  $\varepsilon$  must be found. This bias term is determined by identifying the foot's reported velocity when stationary, which is signified by time  $T$ . When the bias term is subtracted from (2.3), this results in

$$v_a(t) = v_c(t) - \frac{v_c(T)}{T}t, \quad t = [0, T]. \quad (2.4)$$

Processing (2.4) results in a software recalibration, which is completed with each step and markedly reduces position error [11].

In [11], gyroscope data was found to work better than accelerometer data for gait-phase detection. The algorithm detects the zero-velocity stance phase when the combined gyroscope rate falls below an experimentally determined threshold. Zero-velocity stance-phase detection has been an area of focus for several researchers; specifically which sensor or combination of sensors should be used and which type of algorithm should be incorporated to detect the zero-velocity stance phase. Fourati uses both gyroscope and accelerometer data for gait-phase detection, claiming the combination of these sensors more accurately determines the stationary foot, which allows a more precise application of the ZUPT algorithm [49]. His gait-phase detection is accomplished by finding the variance of the squared norms from five to ten samples of data. If the variance is below the experimental threshold, the detector reports stance; if above, swing. By contrast, Swedish researchers find gyroscope data alone to be better than both accelerometer data or combined gyroscope and accelerometer data for zero-velocity detection [50]. Another approach was taken by the MapCraft team, who developed a computational- and energy-efficient method to find user position within a preloaded map using only accelerometers and magnetometers [51]. The team used a map-matching technique employing conditional random fields to determine location within a surveyed building. The disadvantage of this technique is apparent: there are few scenarios in which surveying a building beforehand is practical. Another group used subtractive clustering to determine zero velocity based on sensor-data patterns instead of the conventional magnitudes. This increases the robustness of the gait phase, allowing the user to walk across uneven terrain such as hills [52]. Several other methods have been proposed, but most appear to focus on a single sensor type such as gyroscope or

accelerometer to determine zero velocity. A few researchers use a combination of the typical two- or three-sensor types available on an IMU (e.g., gyroscope, accelerometer, and magnetometer) to detect when the foot is in stance phase. The wide variety of zero-velocity detection methods, along with their unique performance characteristics, suggests that a diverse set of personal navigation systems require an equally diverse repertoire of ways to determine a stance phase. The selection of gyroscopes, accelerometers, and magnetometers available, and their performance, is the driving force in the choice of zero-velocity detection method for a given application.

Some researchers have achieved very good results by mounting a sonar device on both shoes to constrain heading error in addition to using stance phase zero-velocity updates [53]. Such a transmitting sensor is not battlefield compatible because the transmitted signal may alert the enemy to the presence of equipped soldiers, destroying the element of surprise, and any transmission introduces a vulnerability by which the adversary may develop countermeasures to disrupt or disable the capability.

One promising investigation is the incorporation of multiple IMUs to reduce sensor errors. One team used a range constraining foot-to-foot maximum value to more effectively fuse two sensors, one on each foot, into a Kalman filter. This coupling of sensors as well as bounding their measurement to a realistic value provides a more accurate solution than can be achieved otherwise [54].

Another team looked at three filtering approaches. The first consists of using a virtual IMU, where multiple IMUs have their raw outputs mapped to a common reference frame before all measurements are processed through one large Kalman filter. The second is a stacked filter that combines all the raw IMU outputs into a Kalman filter, as in the virtual-IMU approach but additional relative information between IMUs is included to provide better updates. The third is a federated filter that processes each IMU's raw sensor data through its own Kalman filter, then sends the filtered data to one large master filter, which in turn sends the doubly processed data back to the individual filters for further processing. The researchers found that the federated setup does not improve accuracy, whereas the virtual IMU and stacked approaches show modest improvements.

In addition, the researchers found a slight increase in accuracy for each additional IMU above the baseline of two [55].

Finally, a third team that co-located three identical IMUs into a sensor array was able to reduce the bias-drift error from  $35^{\circ}/\text{hr}$  per sensor to a synergistic output of  $0.53^{\circ}/\text{hr}$  after combining the three IMUs into one virtual IMU, with two levels of filtering [56].

None of this research uses a complementary filter comparable to that developed in [11] and employed in this thesis; most rely on a Kalman-filter type of optimal estimator.

### III. EXPERIMENTS

A discussion of six topics and test setups with corresponding results is contained in this chapter. In the first section, the noise performance of the 3-Space sensor module is characterized. The complementary filter, which calculates the user's position from IMU measurements, is discussed in the second, and the tuning process used to optimize the performance of the complementary filter with the 3-Space IMU is detailed in the third. These topics largely repeat the tests in [11] in order to compare a relatively modern IMU manufactured in 2014 with one from 2006.

The relatively unexplored question of optimal sensor placement and strategies in sensor distribution, fusion, and processing are investigated in the remaining sections. YEI 3-Space Data-Logging v2.0 sensor modules were used for all tests.

#### A. THE CHARACTERIZATION OF SENSOR ERRORS

To characterize the 3-Space IMU's errors, static runs were performed and analyzed. Each module's was configured to output raw measurements for the 3-Space modules triaxial gyroscopes, accelerometers, and magnetometers. A Kalman-filtered output was not used because it was thought that doing so might obscure true error values.

##### 1. Test Setup

Sample frequency  $f_s$  was set to automatic, which allowed individual 3-Space sensors modules to dynamically determine the sample rate used during testing. The sample rate tended to remain at approximately 63 Hz in both the static and walking tests, although at times it momentarily dropped to approximately 25 Hz. Although higher sampling frequencies were found to provide better results in [11], any rate over 50 Hz was subjectively deemed adequate for the PNS's trapezoidal integration method. To analyze this assumption further, the shortest duration of a walking event signified by  $t_c$  was found to be a heel strike, which only lasted around 0.1 s (i.e., 10 Hz) [11]; therefore, the minimum sample rate  $f_s$  as defined by the Nyquist sampling theorem,  $f_s > 2t_c$ ,

yields  $f_s > 20$  Hz. Experience shows that sampling physical systems considerably higher than the Nyquist rate may allow more detailed analysis and help with deficiency findings and resolution. With this in mind, a 63 Hz real-world sample rate proved adequate for the walking applications in this thesis. Brief experiments were also conducted while running, and the 63 Hz sample frequency proved adequate in this case as well.

To start sensor characterizations, four of the 3-Space sensors modules were placed atop several layers of packing foam to insulate the IMUs from vibration. This configuration was then set on a wooden desk in the fifth-floor controls laboratory, positioned as far as possible from ferrous or magnetic materials. The sensors were spaced about a half meter apart to avoid magnetic cross-contamination, which might otherwise have been generated by current flowing through each IMU's circuitry. Before data recording began, all sensors received a fresh calibration using the 3-Space sphere-calibration wizard for the accelerometers and magnetometers. Though there is no distinct notification by the software when calibration is complete, the 3-Space sensor suite manual suggests obtaining an estimated density level of 30–50 [29]. The density level corresponds to the coverage of the accelerometer and magnetometer's orientation experienced during calibration. In the case of the 3-Space Sensor Suite, a better calibration is associated with a lower density number.

To calibrate, the IMU was held in the tester's hand, which was twisted and rotated for a few minutes, orienting the IMU in as many directions as possible. The more an IMU was rotated about all axes, the lower the calibration density went, while sensor performance was expected to increase. For test purposes, calibration was considered complete when the estimated sampling density reached a value of 12, as shown in Figure 6. After sphere calibration, gyroscope calibration was performed by laying the IMU on a flat, stationary surface, then selecting the "Calibrate Gyros" button on the 3-Space Sensor Suite GUI. Another calibration method, the gradient-descent calibration wizard, is available in the 3-Space sensor suite, but early experimentation found it laborious and without noticeable improvement over the simpler sphere-calibration wizard used exclusively in this research.

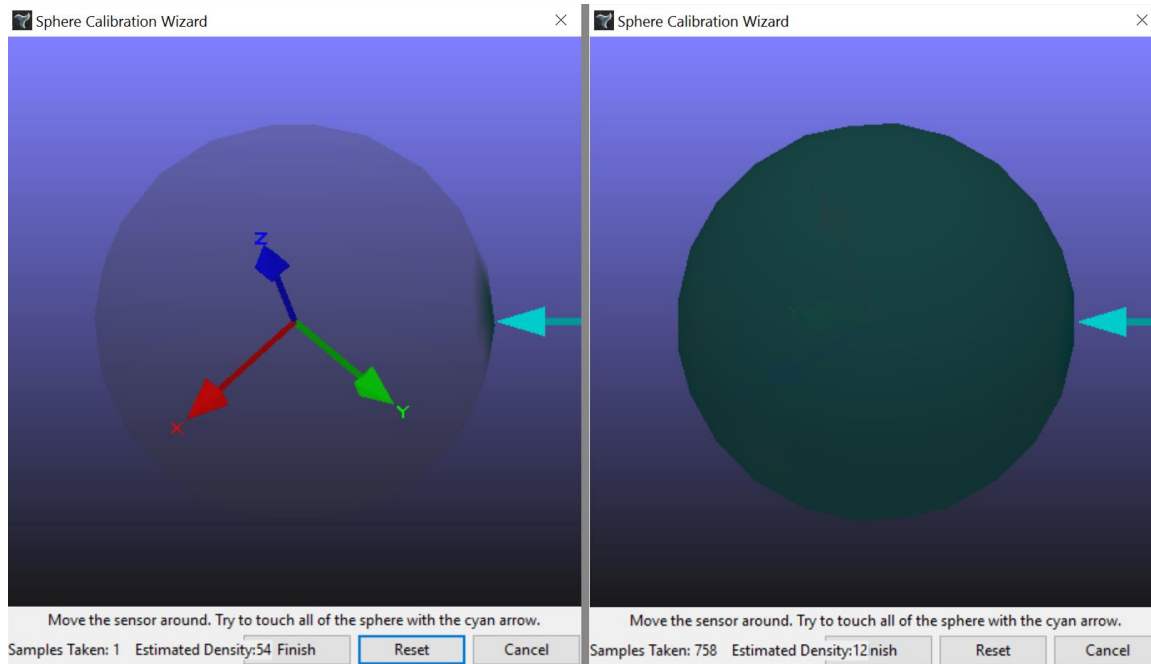


Figure 6. Sphere Calibration Wizard before (Left) and after (Right) Accelerometer and Magnetometer Calibration

To begin static testing, the sensors were turned on and allowed to run uninterrupted for 1 hour, 40 minutes. This period was chosen because it was approximately the same amount of time [11] used for sensor characterization tests. Four static tests were attempted before a satisfactory test setup was achieved. It was originally thought that long duration static tests were better, so the sensors were left to run overnight. Unfortunately, each IMU only contained about three hours of useful battery life. Two of the sensors failed to save collected data before powering off. This difficulty was avoided by limiting data collection to 1 hour, 40 minutes, with a secondary justification that it would allow direct comparison to [11]’s static tests. Satisfactory results were achieved by doing so.

To begin analysis, sensor data were imported into MATLAB. Sample data collected from one of the properly saved three-hour tests were used with an author-created MATLAB GUI program designed to import 3-Space sensor data TXT files and convert them into an HDF5-format-based MATLAB (MAT) file [57]. This tool worked well with ten minutes of sample data. In attempting to process three hours of test

data, however, the program halted after using all 16 GB of onboard random-access memory (RAM). To address the memory issue, a more memory-efficient import script was developed by adapting code from [10]. Creating code to import 3-Space sensor data presented a challenge because the sensor module wrote data to a TXT file that contained multiple data capture formats and used inconsistent methods to separate variables; data formats might be in a date/time or numerical arrangement, while variable separation was signified by a space in some columns and a comma in others. Besides providing efficient code to import data, the author reduced recording time from three hours to just over 1 hour, 35 minutes to match [11]. The complete code, entitled “IMPORT LARGE YEI 3-SPACE TEXT FILES INTO MATLAB,” is provided in Appendix A.

A second set of tools was created to produce a PSD of each sensor’s raw output from the static run. The PSD is used as a visual aid to convert the output of a motionless sensor from the time to the frequency domain. This exposes characteristics of the signal that would otherwise remain hidden [58]. The PSD reveals power created at specific frequency components of a signal. For an IMU feeling no force, the PSD uncovers noise power. Noise power increases due to the random error components of thermal, shot, and flicker noise [59]. The integral of the PSD is the sensor’s noise power. Noise power is the source from which angular random walk specifications are derived and can be useful because they indicate a sensor’s noise performance. A lower ARW number suggests a lower standard deviation from measurement data error. Higher-quality sensors are expected to provide a noise-power output level below that seen with a lower-quality sensor. When two sensor modules of different quality are compared, the better sensor will list a smaller ARW specification on its datasheet.

## **2. PSD of Sensor Outputs**

The PSDs were created using MATLAB’s version of the Welch power spectral-density estimate function *pwelch*( ), which implements Welch’s time averaging using a short-periodogram method [60]. The Welch method uses an efficient fast Fourier transformation (FFT) to identify those frequencies that hold more power than others [61]. To display plot trends that are more visually apparent, the FFT’s windowing parameter

was adjusted with the MATLAB function *triang()* and set to execute 500 samples for each section. The Welch function computes many small FFTs along the user-specified sliding window size and averages the magnitude-square value for each. This prevents noisy measurements from obscuring data trends and small power spikes and smooths the plots, rendering them more visually intelligible. The *pwelch()* function proved ideal in facilitating noise-performance comparisons among sensors of the same type.

To demonstrate how a sensor's noise PSD should appear, a MATLAB simulation was performed with randomly generated, uncorrelated values between zero and 0.006 as created by the *rand()* function. The latter value was chosen to reasonably mimic noise generated from a stationary gyroscope. The resulting large set of random data was then processed with the *pwelch()* function using MATLAB default values except for a sampling frequency  $f_s$  setting of 63 Hz. This value is familiar as the sample frequency of the 3-Space module when the sampling interval was set to automatic. The resulting PSD, presented as the upper plot in Figure 7, is very noisy.

In an attempt to smooth the PSD for easier interpretation, the original raw data were reprocessed through *pwelch()*, but this time the time-function parameters were adjusted from the MATLAB defaults to new empirically found settings. The smoothed PSD is displayed as the bottom plot in Figure 7. Note that the dc (i.e., 0 Hz) power component of the smoothened PSD at the bottom of the figure bleeds slightly into higher frequencies. This bleed-over is an artifact produced by the function parameters used for *pwelch()*. This should be remembered later when explaining large power components in the lower frequencies (i.e., less than 2.0 Hz) of subsequent PSDs. To aid in comparing PSDs, the mean  $\mu$ , standard deviation  $\sigma$ , number of samples, and sample frequency  $f_s$  of the MATLAB-generated data were calculated and displayed in Figures 7–13.

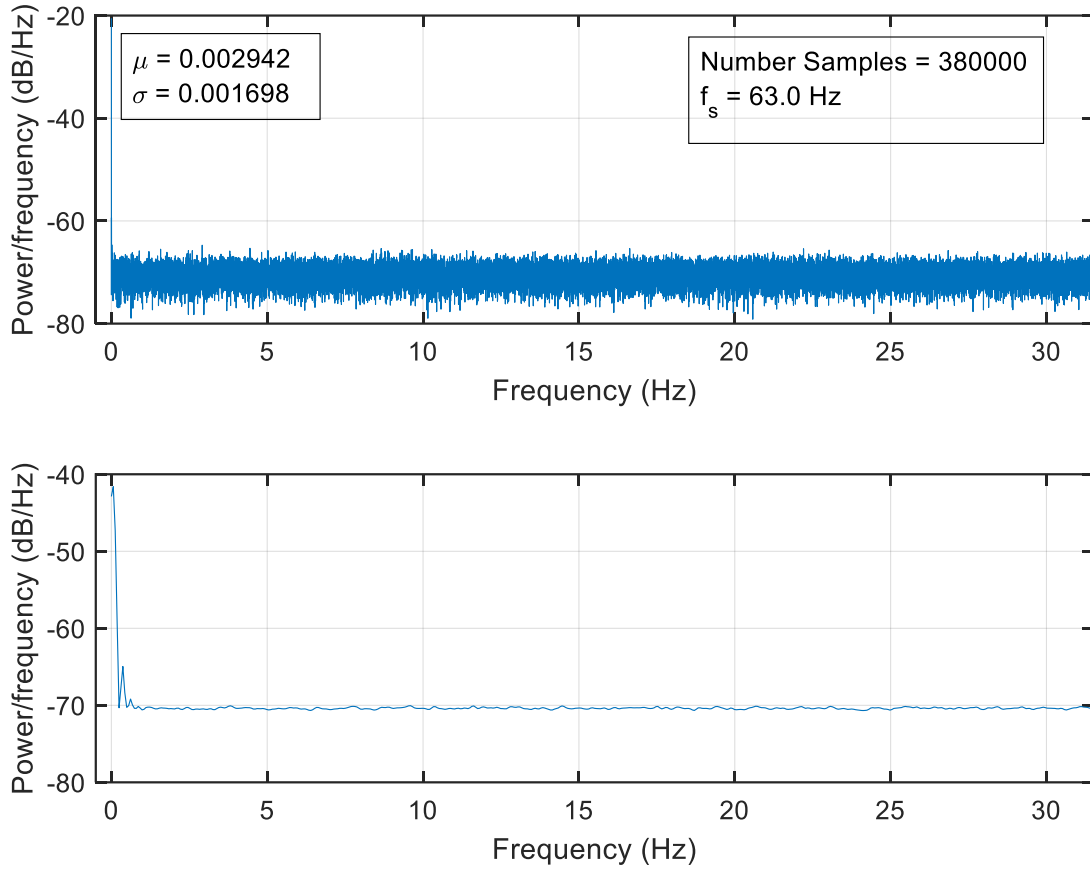


Figure 7. Simulated PSD of Randomly Generated Noise (Top), and Same Data Processed using Welch PSD (Bottom)

Given the baseline PSD containing MATLAB-generated random data, additional PSDs were derived from real sensor measurements and presented, starting with the gyroscope. Of all sensor types, the gyroscope alone should sense zero forces while stationary on a foam pad. This is different than an accelerometer, which always feels the force of Earth’s gravity, or the magnetometer, which always senses a magnetic field.

The first experimentally derived PSD of gyroscope data is provided in Figure 8. The data underlying the PSD were taken from a single 3-Space module designated as Sensor 1. There are three lines of data because each sensor type in a 3-Space sensor module is triaxially configured—that is, contains three identical, orthogonally arranged gyroscopes to measure forces in three axes. In the interest of reducing clutter, the mean

and standard deviation from each of the three gyroscopes within a triaxial configuration are combined.

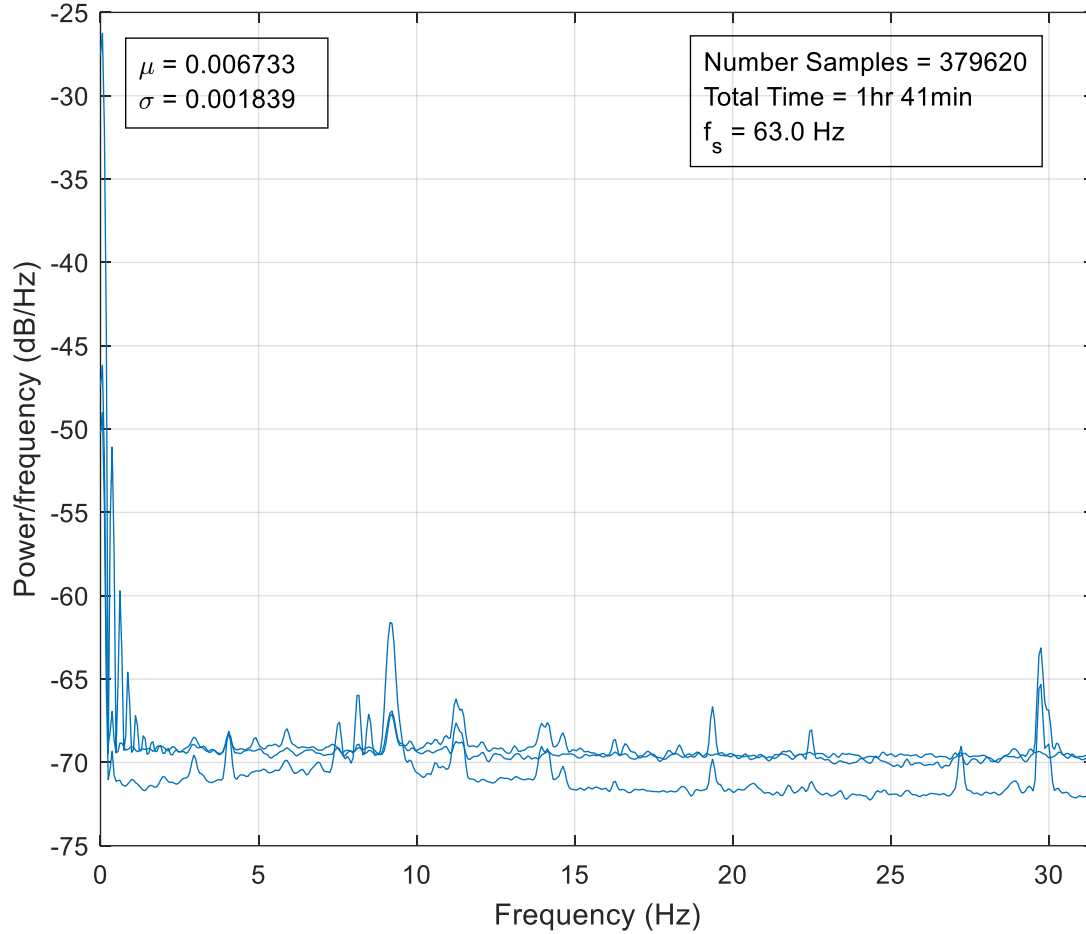


Figure 8. PSD of Sensor 1's Raw Triaxial Gyroscope Outputs

For most frequencies, all three gyroscopes from Sensor 1 display noise levels near  $-70$  dB. With the noise-power level identified, the 3-Space module, released in 2014, is compared with the MicroStrain 3DM-GX1, released in 2006. Using reported information from [11], we have the noise of the MicroStrain 3DM-GX1 as  $-65$  dB. This indicates the older IMU's noise performance is  $5.0$  dB worse than the current unit. That is, the gyroscope within a contemporary AHRS IMU that sells for  $\sim \$250$  had  $-5$  dB better noise performance than an AHRS IMU that sold for  $\sim \$3,000$  almost a decade before. The real-world effect of this performance differential should be reflected by the 3-Space

sensor having a lower ARW (i.e., standard deviation) than that of the MicroStrain 3DM-GX1. This prediction agrees with the ARW specifications in Table 1, where the ARW for the MicroStrain 3DM-GX1 sensor is listed as  $3.5^{\circ}/\sqrt{\text{hr}}$  and  $0.45^{\circ}/\sqrt{\text{hr}}$  for the YEI 3-Space Sensor v2.0.

The noise performance of Sensor 1 is better than the older sensor's but with noticeable power spikes above the noise floor. These divergences consisted of a large dc power contribution and smaller power spikes close to 9 Hz and 30 Hz. Note that if the default values with regard to the *pwelch*( ) function were used for the parameters “window,” “nooverlap,” and “nfft,” then the plot would correctly show the dc noise bias at 0 Hz instead of the additional power displayed near 1 Hz. The dc component is reasonably explained as a result of characteristically white data, a shared issue with the simulated data at the bottom of Figure 7; however, the power spikes at 9 Hz and 30 Hz are not explained. No power spikes were anticipated above the noise floor because the data were taken from stationary gyroscopes; the PSD is expected to show a near-constant power level throughout the frequency band. The presence of power at multiple frequencies is interpreted as evidence of infiltration by another power source, possibly resonance picked up during the test run from the many other operations within the five-story building or from the half-dozen personnel who walk in and out of the laboratory closing doors behind them, triggering reverberations that are felt in nearby walls and make their way to the floor, up the table legs, through the foam, and into the IMU. The resulting oscillatory energy might consist of 9 Hz and 30 Hz components, since that is where most of the power spike energy is plotted.

The mean and standard deviation of Sensor 1's gyroscope output is displayed in Figure 8. Before processing data through the *pwelch*( ) function, raw data from Sensor 1's triaxial arrangement of gyroscopes were combined to find the resultant mean power level and standard deviation. The gyroscopes can display negative numbers, so the non-zero mean might indicate a gyroscope bias error in the raw output. A non-zero mean was purposefully created for the earlier simulation shown in Figure 7, as random numbers were distributed between zero and 0.006. Rather than create an individual PSD plot for every 3-Space module, the author plotted all gyroscope data from the four sensor

modules onto the same PSD to provide a better example of power-frequency trends. The data from the compiled PSDs are exhibited in Figure 9. This compilation contains the data from a total of 12 gyroscopes, extracted from the four sets of triaxial gyroscopes. The IMUs recorded data simultaneously, and all were placed on the same foam pads. As the data shown in Figure 9 demonstrates, noise levels differed slightly among the individual gyroscope sensors within the same 3-Space module and as compared to other IMUs (i.e., sensors 2, 3, and 4). The data shows that the 9-Hz power spike was not relegated to Sensor 1 but affected all sensors. Additional power spikes were more recognizable and clustered near 12 Hz, 14 Hz, 19 Hz, 22 Hz, 27 Hz, and 30 Hz. Again, *pwelch*( ) appears to move some power toward 1 Hz, but the unadjusted noise plots show that power was actually a dc component.

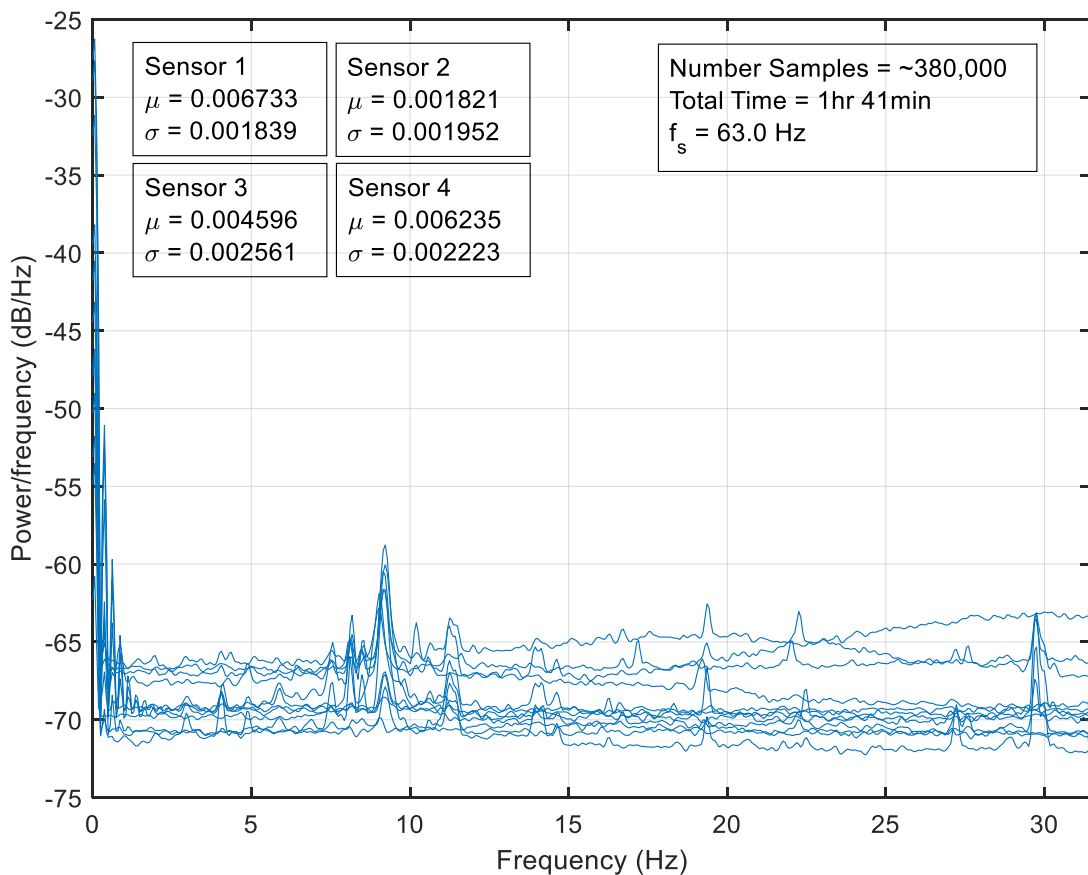


Figure 9. PSD of All Four IMUs' Raw Triaxial Gyroscope Outputs

The similarities among discrete gyroscopes may reflect the fact that they recorded data at the same time and under the same conditions. The differences among sensors as to mean and standard deviation suggest that the magnitudes of their error biases were different. The variety of noise performances among identical sensors exemplifies the difficulties within the world of MEMS manufacturing.

A characterization of the accelerometer PSD was also performed, as presented in Figures 10 and 11. It was found that the effects of gravity could not be removed, creating significant power below 7 Hz for one axis of the triaxial accelerometer. It is assumed that this accelerometer was oriented to measure the gravity vector, but it appears that the other accelerometers measured a small gravity component as well. The combined mean was also skewed by gravity. It should be noted that the displayed statistics from each sensor within the triaxial accelerometer were combined to reduce clutter.

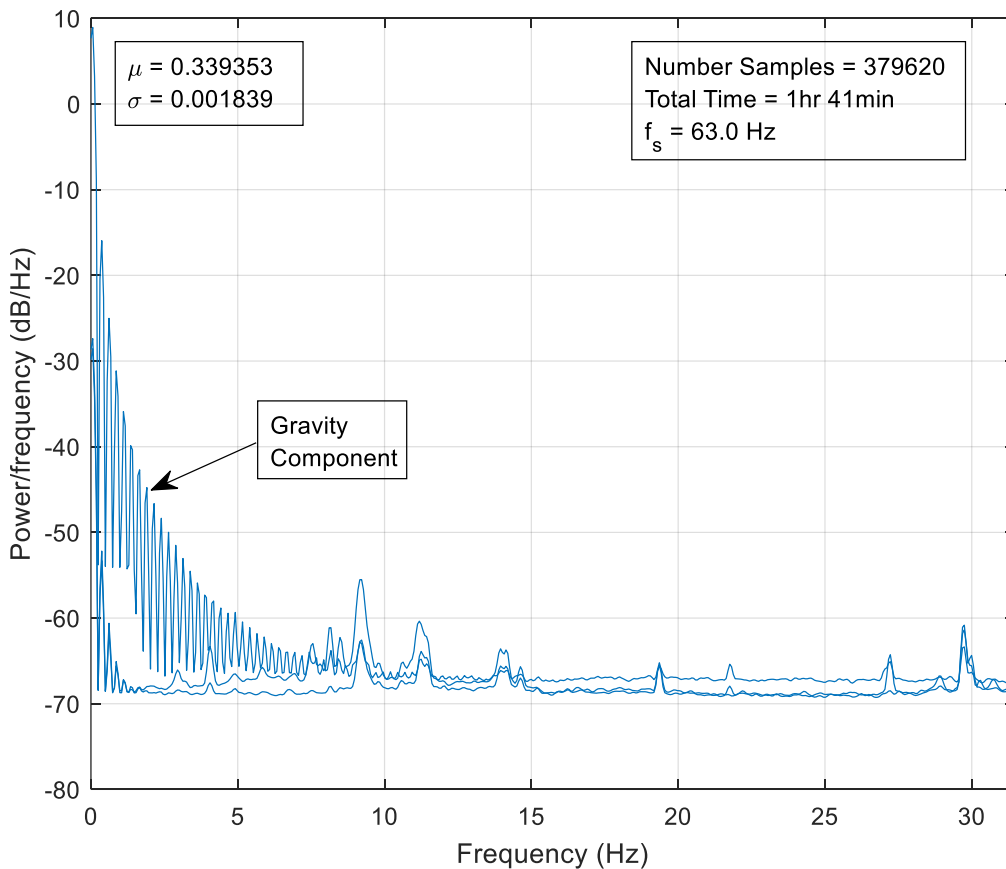


Figure 10. PSD of Sensor 1's Raw Triaxial Accelerometer Outputs

Combining all 12 accelerometers from the four IMUs onto one plot reveals data trends as displayed in Figure 11. Performance among the four sensors was very similar, with power spikes occurring at the same frequencies as seen with the gyros. Again, the likely explanation is that some outside source stimulated both the linear acceleration and angular velocity-measuring devices at the same frequency.

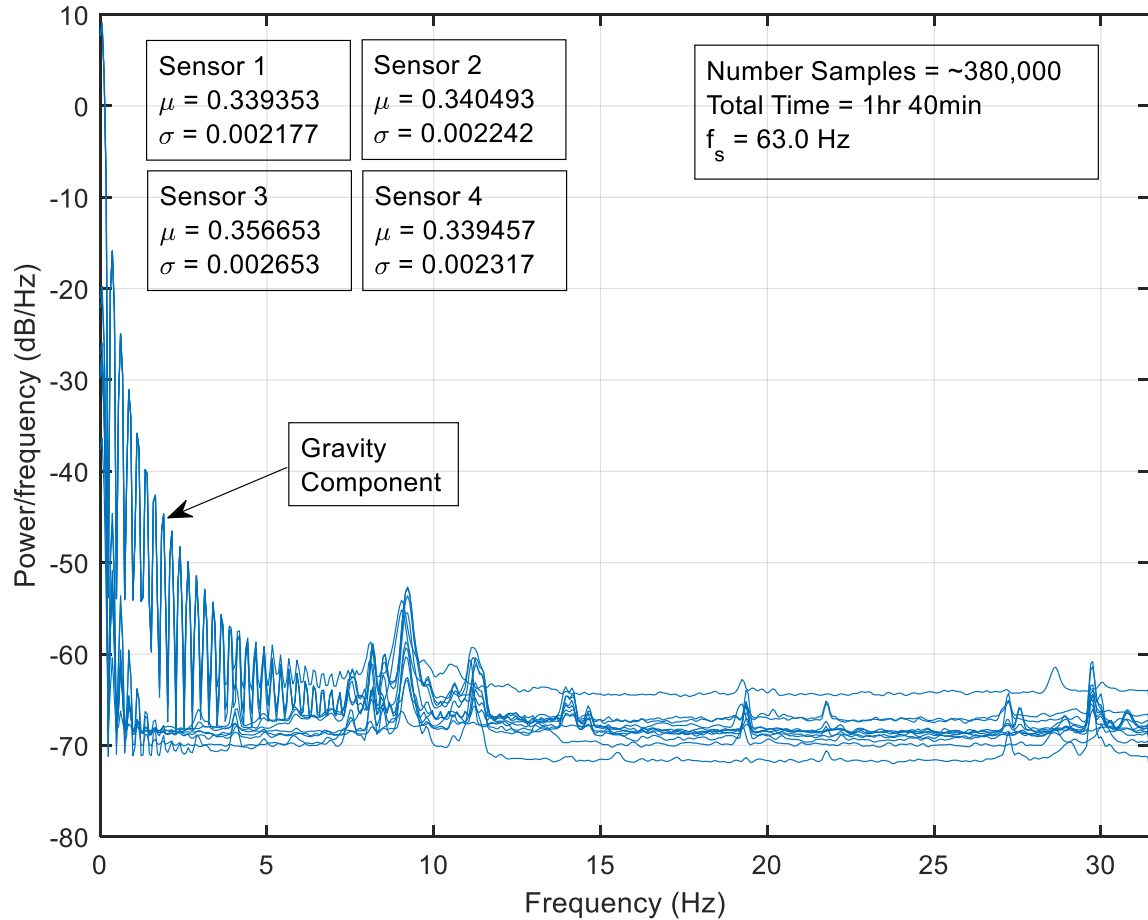


Figure 11. PSD of All Four IMUs' Raw Triaxial Accelerometer Outputs

Next, the PSD of the magnetometers was found. The PSD for Sensor 1's three magnetometer outputs is shown in Figure 12, and a PSD plot containing all 12 magnetometer outputs in Figure 13. These outputs are nearly indistinguishable. Unlike the gyroscope and accelerometer measurements, there were no power spikes, indicating a near-constant magnetic force. The linear and rotational forces, which resonated through

the accelerometers and gyroscopes, were not felt by the magnetometers. Like the previous PSDs, the displayed statistics from each triaxial sensor were combined to reduce clutter.

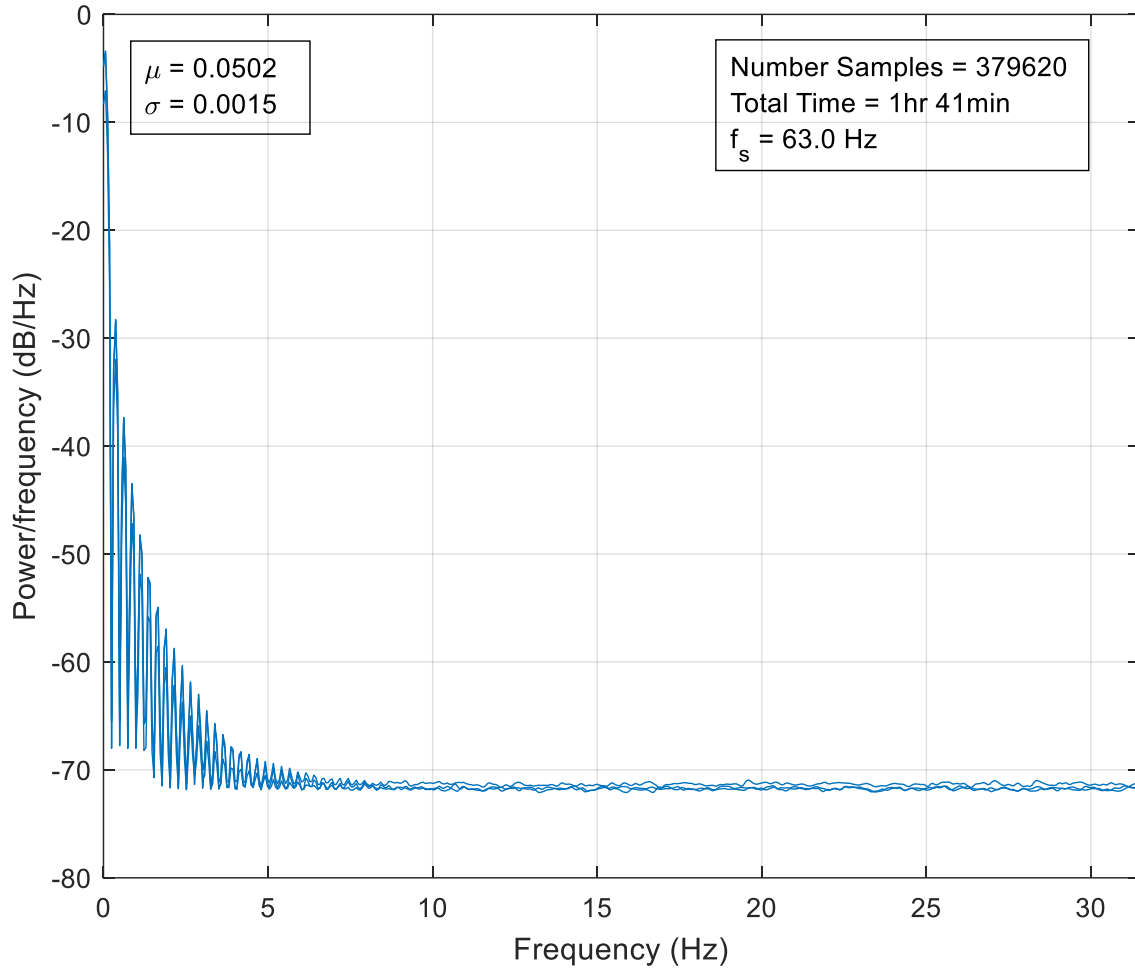


Figure 12. PSD of Sensor 1's Raw Triaxial Magnetometer Outputs

In Figure 12 and Figure 13, the magnetometer data settles down to about  $-71$  dB as frequency increases. This noise level is slightly less and significantly more consistent than the accelerometer and gyroscope data. The reasons behind this reduced noise are unknown. Perhaps the phenomenon that caused power spikes in the gyroscope and accelerometer PSD plots did not contain a pulsating magnetic field, but a near constant one.

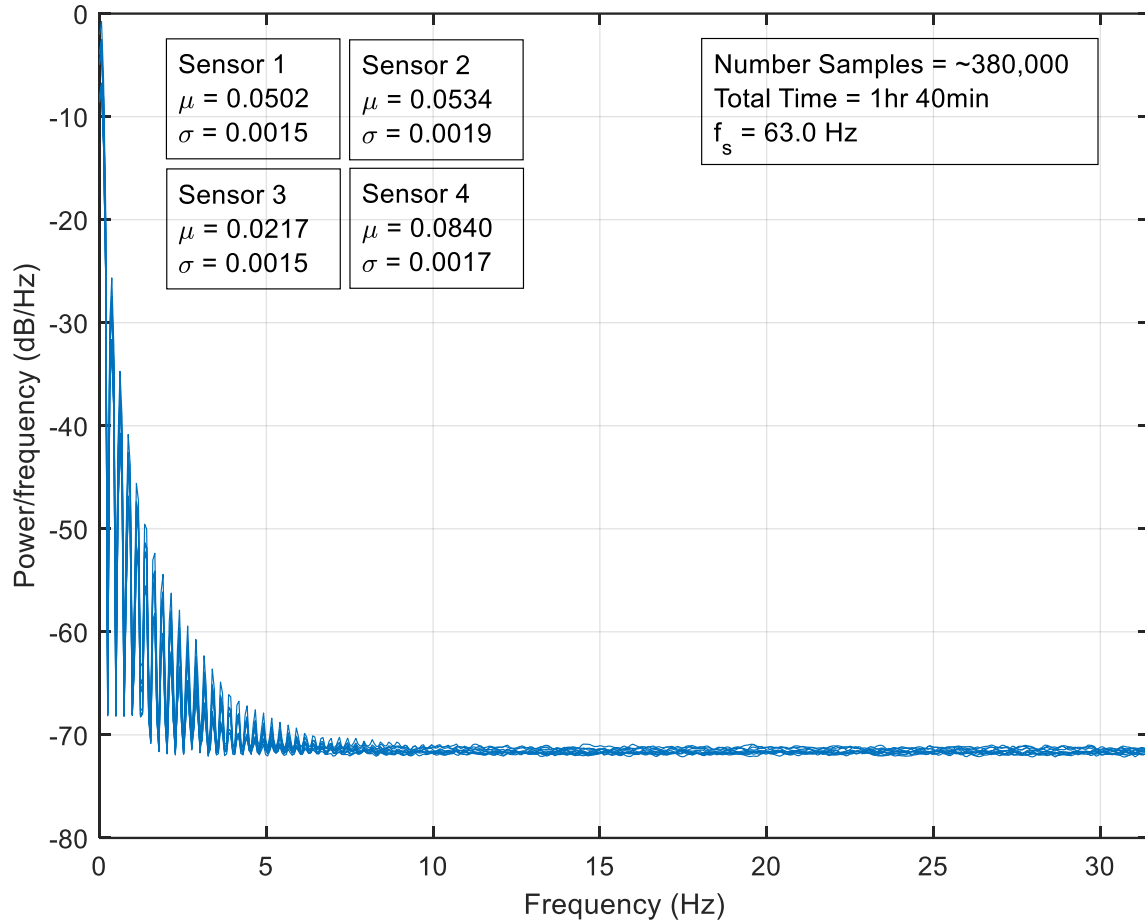


Figure 13. PSD of All Four IMUs' Raw Triaxial Magnetometer Outputs

### 3. Histogram of Sensors

To characterize sensor bias, a histogram for Sensor 1 was completed using the same static-run data as the noise PSDs. Because the other three sensors had characteristics similar to those of Sensor 1, they were not plotted. If Sensor 1 showed no bias error, a histogram containing the static-run measurements from its three gyroscopes would have a zero-centered mean; however, as indicated by the three different colors representing the triaxial gyroscope data in Figure 14, Sensor 1 did display bias error, the magnitude of which varied among gyroscopes. For unknown reasons, Gyroscope 1 had the largest bias error within Sensor 1. This undesirable bias could not be explained out as the rate of Earth's rotation, which is much smaller in magnitude at only 0.0042 deg/s. The bias error also illustrates the need for a ZUPT algorithm, which can remove Gyroscope

1's bias error. The corresponding histogram for each gyroscope is characterized by a Gaussian distribution, which may indicate a wide variety of error sources whose standard deviation is the ARW. Overlapping data on the left set of histograms in Figure 14 are represented by third color not seen in the legend.

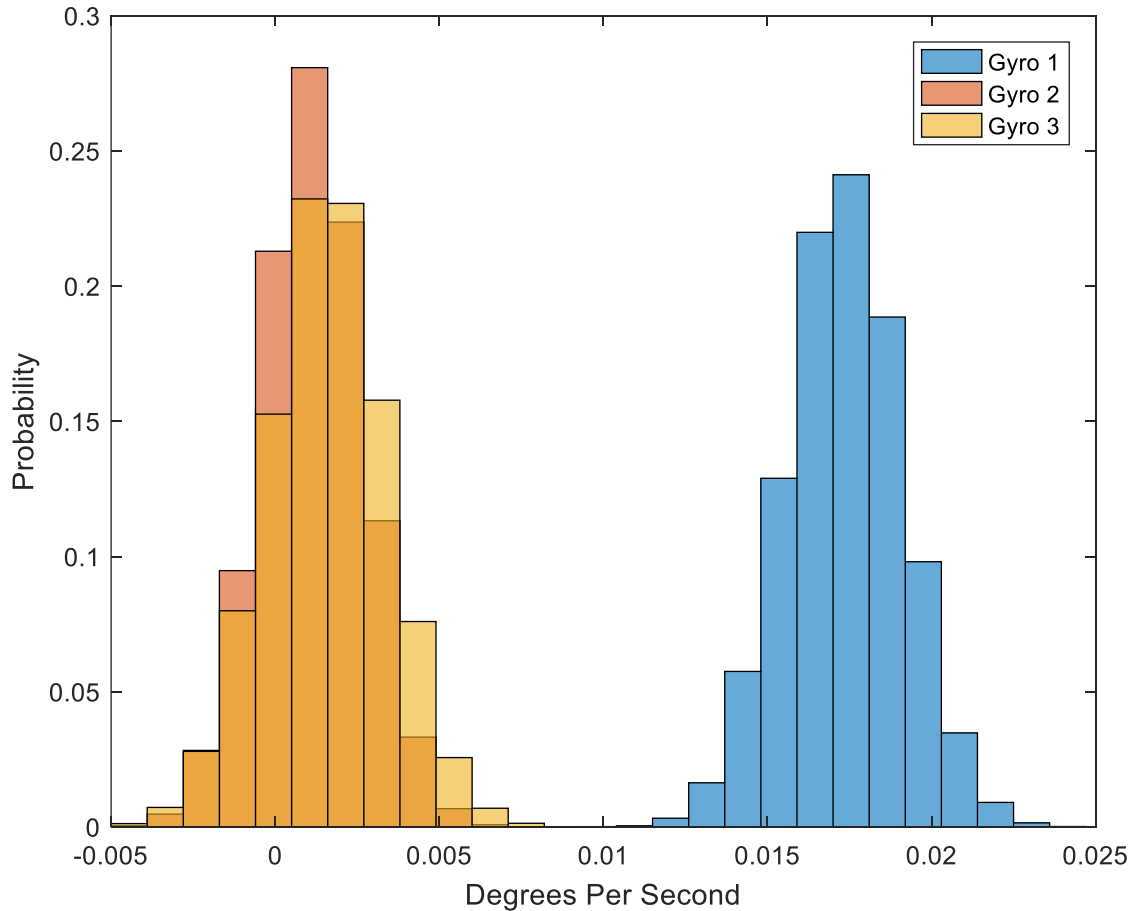


Figure 14. Histogram of Sensor 1's Raw Triaxial Gyroscope Outputs

Sensor 1's accelerometers also had a near-Gaussian distribution, as displayed in Figure 15. While it appears that all three of Sensor 1's accelerometers were plotted, the histogram actually shows only two. The X-axis accelerometer, which felt most of the gravitational force, is not shown, because the scale of the resulting plot made each histogram indistinguishably small. The appearance of different color intensities signifies data overlap. As noted, all the accelerometers appear to sense a small gravity component, despite orienting the Y-Z plane perpendicular to the gravity vector. For the data shown in

Figure 15, the *Y*-axis accelerometer data are on the right, and the *Z*-axis accelerometer data are on the left. Overlapping data in Figure 15 is represented by a third color not seen in legend.

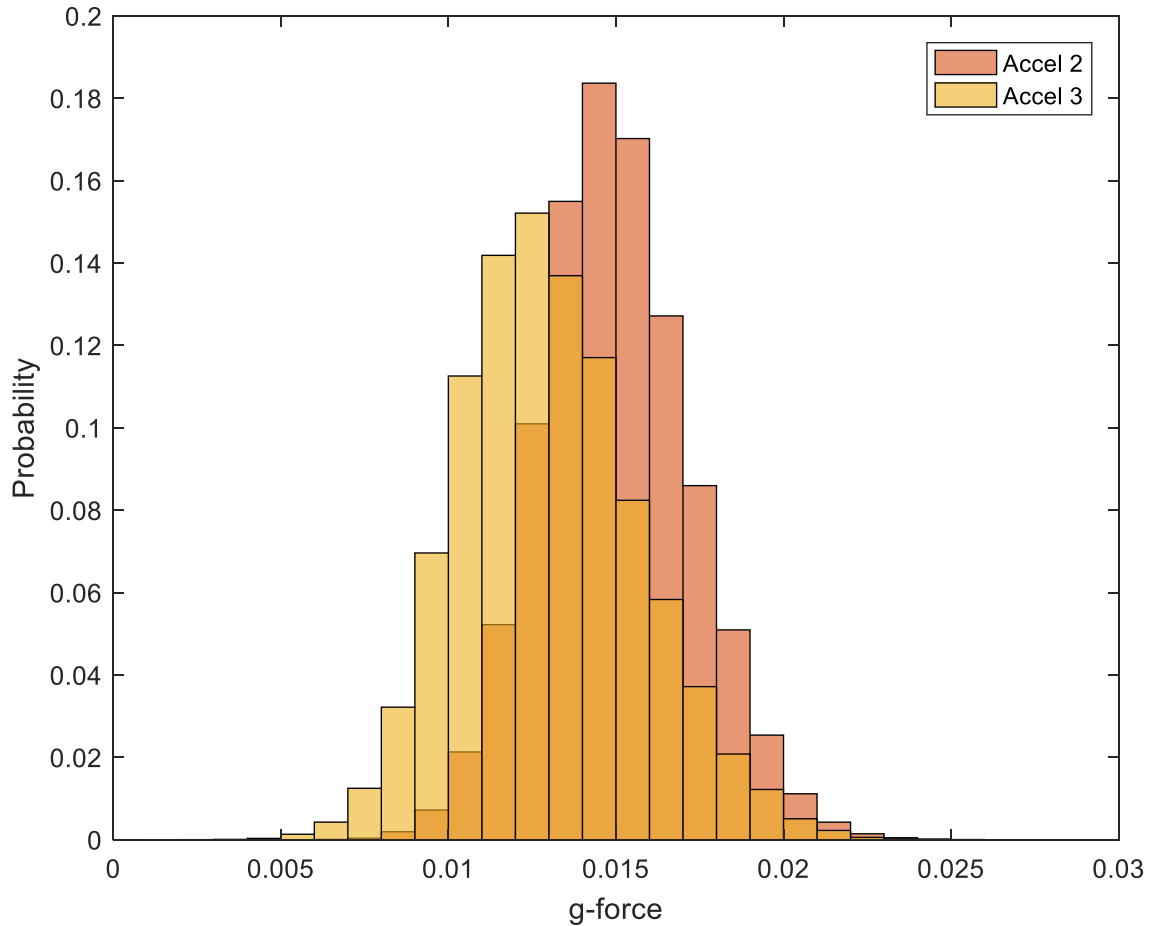


Figure 15. Histogram with Two of Sensor 1's Raw Accelerometer Outputs

As with the two previous sensor types examined, a histogram was made of all three magnetometer measurements from Sensor 1's static run, as presented in Figure 16. Like the accelerometers, each magnetometer had a different mean magnitude, so it was impossible to display them clearly on the same histogram; therefore, only the *Z*- and *Y*-axis sensors are shown, which appear to have a Gaussian distribution like the other sensor types. For some reason, the data distribution of the *Z*-axis magnetometer (left) is a mirror image of the *Y*-axis magnetometer (right). This symmetry, as well as a much lower

spread of data than seen in the other sensor types, may indicate more consistent measurements, or perhaps magnetometer noise is lower than gyroscope and accelerometer. Another possibility resides in the power spikes seen at multiple frequencies in the gyroscope and accelerometer PSDs. The power in several of these frequencies may have been a source of noise, increasing the standard deviation and data spread on their histograms. The magnetometer was not affected by power spikes in its PSD, so noise levels were reduced, resulting in a lower standard deviation and data spread in the histogram.

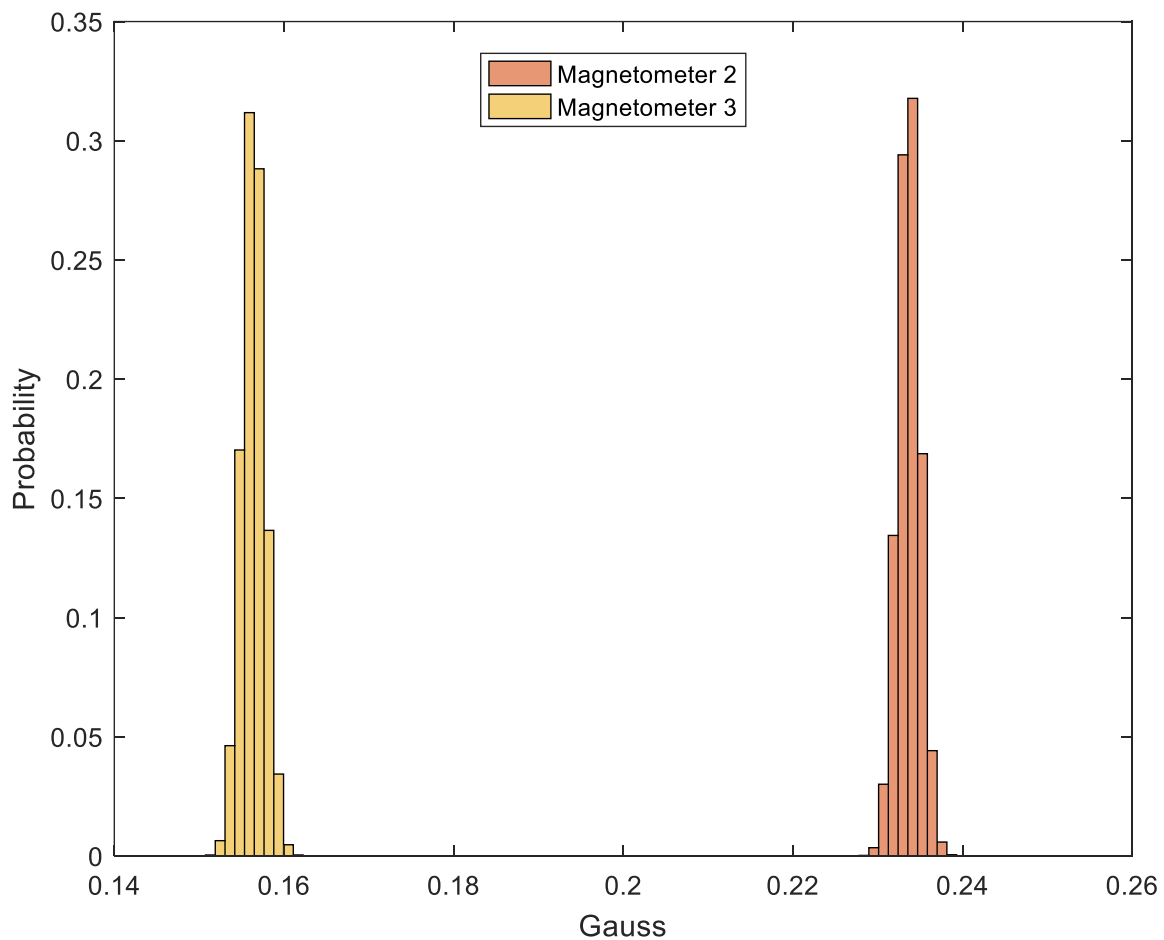


Figure 16. Histogram with Two of Sensor 1's Magnetometer Outputs

## **B. THE COMPLEMENTARY FILTER**

The quaternion-based complementary filter employed in this research processes measurement data from multiple sensor types into a blended solution. This filter was split into two parts: the factored quaternion algorithm (FQA), used for the stance phase of walking, and angular-rate measurements, used for the swing phase. The FQA uses accelerometers to determine pitch and roll, while the magnetometer detects yaw. The complementary filter synergistically combines the FQA with angular-rate data to output a quaternion estimate, which is better than the FQA alone [11].

### **1. Reference Frames**

Before discussing the complementary filter, it is important to provide the reference frames used by the PNS. Since the dynamics of human movement are very low relative to air- and spacecraft, a flat-earth model may be used with a low-cost IMU without significant loss of accuracy. Such a model assumes the Earth's radius to be infinite, therefore, flat and free of any Coriolis effect due to rotation. These assumptions enable a human-mounted IMU to describe orientation in terms of roll, pitch, and heading with respect to a local coordinate system [62]. In the PNS, the local coordinate system is north-east-down (NED). This system is fixed to the Earth's surface, with its starting point typically given as the location where the IMU starts recording. For this research, the IMU's  $X$ ,  $Y$ , and  $Z$  axes correspond to north, east, and down, respectively.

The term "reference frame" is often used in discussing a local coordinate system, but this is incorrect. A reference frame is a tool used to describe motion. When walking, inertial forces, that is to say, linear and angular forces, are felt depending on the walker's movement with respect to the earth's surface. A coordinate system is then needed to determine where he or she is located terrestrially at a given time. Simply put, a coordinate system details where exactly an individual's position is within a reference frame.

A body frame is the frame whose orientation is fixed with respect to the sensor module. It places the center of the IMU coordinate system at its centroid. The 3-Space module body frame consists of  $X$ -,  $Y$ -, and  $Z$ -axes whose orientations remains fixed to the IMU as it rotates. The 3-Space sensor module body axes in Figure 3 are arbitrarily

configured. The  $X$ -axis extends from the top of the sensor module, the  $Y$ -axis extends from the left side when the module's buttons are facing the user, and the  $Z$ -axis exits the bottom where the micro-USB plug attaches to the IMU. Since all forces are perceived by the IMU with respect to its body frame, these measured forces are translated to a common, local, NED coordinate system to provide usable location information.

## 2. Factored Quaternion Algorithm: Stance Phase

The FQA uses measurements from the 3-Space module's triaxial accelerometers and magnetometers, which provide roll, pitch, and heading (yaw) information in the stance phase. Per design, the gyroscope was left out of FQA calculations since the FQA was configured to work with angular position measurements (i.e., degrees) as opposed to angular rates (i.e., deg/s).

As a quaternion-based system, the FQA offers robust performance and prevents “gimbal lock,” a disruption that occurs when one or more Euler angles reach an attitude of  $90^\circ$ . At this inclination, the mathematics of electronic gyroscopes break down and can result in a singularity, resulting in disruptions to software code with divide-by-zero events. Quaternions avoid singularities altogether and provide good internal orientation.

A quaternion is made up of a scalar  $q_0$  and a vector  $\mathbf{q}$

$$q = q_0 + \mathbf{q} = q_0 + iq_1 + jq_2 + kq_3 \quad (3.1)$$

where the scalar components of a quaternion are  $q_0, q_1, q_2, q_3$ .

Although this complex vector does not suffer from the singularities seen with Euler angles, quaternion measurements are difficult to intuitively follow. To compensate, the FQA computes quaternions for the benefit of the PNS and performs additional processing to complete Euler-angle transformations for the sake of intelligibility.

The FQA used in this research consists of three separate quaternions, each corresponding with a rotation about the  $X$ -,  $Y$ -, or  $Z$ -axis. The result is that the components in the body frame of the sensor can be converted to a flat-earth model

navigation frame. This combined output provides values for roll, pitch, and heading in the form of a composite quaternion

$$q = q_{Roll} q_{Pitch} q_{Yaw}. \quad (3.2)$$

The accelerometers provide measurement data for the roll and pitch quaternions, while the magnetometer provides data for the heading (yaw) quaternion [11]. Quaternions cannot simply be multiplied together; they are non-commutative, and order is important. Several quaternion-specific formulas are given in [63].

***a. Accelerometer Quaternion: Roll and Pitch***

Since the accelerometer measures roll and pitch primarily in the stance phase, the complementary filter assumes that gravity is the only force measured. The 3-Space module's accelerometer-output units are expressed in units of g-force, where the norm of the gravity vector  $|\vec{a}^b|$  was assumed to equal one. As stated previously, the local g-force in Monterey, California is equal to 9.7991 m/s<sup>2</sup> [35]; therefore, the norm of the g-force vector is  $9.7991/9.80665 = 0.99923g$ . For the purpose of detecting pitch and roll, a difference of 0.08% between the SI value and the local value was deemed negligible. The individual components of the gravity vector are,  $\vec{a}^b = [a_x^b, a_y^b, a_z^b]$ . The superscript  $b$  is used to signify that measurements were taken in the sensors body frame [11].

The roll and pitch quaternions were computed from accelerometer measurements. Roll is represented by  $\phi$ , and the roll quaternion is constructed as

$$q_{Roll} = \begin{bmatrix} \cos \frac{\phi}{2} \\ \sin \frac{\phi}{2} \\ 0 \\ 0 \end{bmatrix}, \quad (3.3)$$

where

$$\sin \phi = \frac{-a_y^b}{\cos \theta} \quad (3.4)$$

and

$$\cos \phi = \frac{-a_z^b}{\cos \theta} . \quad (3.5)$$

Pitch is represented by  $\theta$ , and the pitch quaternion is configured as

$$q_{Pitch} = \begin{bmatrix} \cos \frac{\theta}{2} \\ 0 \\ \sin \frac{\theta}{2} \\ 0 \end{bmatrix} , \quad (3.6)$$

where

$$\sin \theta = a_x^b \quad (3.7)$$

and

$$\cos \theta = \sqrt{1 - \sin^2 \theta} . \quad (3.8)$$

#### ***b. Magnetometer Quaternion: Heading (Yaw)***

The FQA relies on magnetometer readings to determine heading in a navigational frame. Determination of heading through other sensor technologies, such as accelerometer or gyroscope, is impractical under current MEMS IMU accuracy levels; this magnetometer strategy provides a stopgap. When low-cost, small form-factor gyroscopes reduce their error bias-stability performance specification by at least a magnitude, the magnetometer may be replaced [46].

Magnetometers are not ideal as a heading reference, because ferrous materials and currents in wires may distort the local magnetic field, degrading accuracy. Unfortunately, MEMS gyroscopes such as those on low-cost IMUs are not nearly accurate enough to use as a heading source. To meet PNS performance demands, a magnetometer is currently required.

Heading is determined by feeding magnetometer measurements into a yaw quaternion. With the yaw angle represented by  $\psi$  and the roll and pitch quaternions

represented by  $\begin{bmatrix} N_x & N_y \end{bmatrix}$ , it is possible to determine the horizontal components of the magnetic vector. The magnetic measurement vector is signified by  $\begin{bmatrix} M_x & M_y \end{bmatrix}$ , while the heading equation is [11]

$$q_{Yaw} = \begin{bmatrix} \cos \frac{\psi}{2} \\ 0 \\ 0 \\ \sin \frac{\psi}{2} \end{bmatrix}, \quad (3.9)$$

where

$$\begin{bmatrix} \cos \psi \\ \sin \psi \end{bmatrix} = \begin{bmatrix} M_x & M_y \\ -M_y & M_x \end{bmatrix} \begin{bmatrix} N_x \\ N_y \end{bmatrix}. \quad (3.10)$$

Previous work found the FQA best suited to processing data in the stance phase [11]. This conclusion was reaffirmed through testing, in which an observation was made that the swing-phase accuracy of the PNS improved when the FQA was significantly de-weighted. Accurate gait-phase detection also enabled optimal FQA performance. To achieve fast detections and high accuracy, a balance was needed between allowing the system to detect changes in gait phase quickly and minimizing false detections.

### 3. Angular Rate Measurements: Swing Phase

Positional changes of an IMU-equipped user were primarily found by integrating navigation-frame accelerations. Testing showed the most limiting sensor for the PNS was the gyroscope. Due to poor results when using dead reckoning with MEMS IMUs, the need for a PNS with error-eliminating algorithms was evident.

During the swing phase, attitude was found with the dynamic rate quaternion  $\dot{q}_d$  (i.e., the gyroscope quaternion) using [11]

$$\dot{q}_d(t) = \frac{1}{2} \Omega(\omega) q_d(t) \quad (3.11)$$

where  $\Omega(\omega)$  is the skew-symmetric matrix containing body-frame angular rate measurements  $\omega_{xb}$ ,  $\omega_{yb}$ , and  $\omega_{zb}$ . The angular rate measurements were extracted from the three gyroscopes about their axes using the skew-symmetric matrix

$$\Omega(\omega) = \begin{bmatrix} 0 & -\omega_{xb} & -\omega_{yb} & -\omega_{zb} \\ \omega_{xb} & 0 & -\omega_{zb} & \omega_{yb} \\ \omega_{yb} & \omega_{zb} & 0 & -\omega_{xb} \\ \omega_{zb} & -\omega_{yb} & \omega_{xb} & 0 \end{bmatrix}. \quad (3.12)$$

Though the gyroscope was most actively used during the swing phase, the key to PNS performance and keeping gyroscope error in check is found in the stance phase. The PNS treats gyroscope measurements during the stance phase as an error bias. Since there should be a near-zero velocity of the standing foot, any angular movement measured by the gyroscope is assumed erroneous. The ZUPT algorithm identifies falsely reported angular-rate values and uses them to eliminate error biases from the immediately previous swing-phase measurements. That is, the error biases from a single stance phase are subtracted from a single previous swing phase, and the gyroscope is effectively recalibrated every time the instrumented foot hits ground.

#### 4. Gait-Phase Detection

Gait-phase detection is used by the PNS to determine the proper gain needed to achieve optimum performance and accuracy. An angular rate threshold was empirically found and set at a value that allowed the PNS to determine if the user was in a stance or swing phase. This angular rate threshold can be considered a low-pass filter; if below the threshold, the FQA is allowed to pass, if above, the FQA is attenuated but still present. An example is provided in Figure 17, where the angular rate threshold was set to 0.35 deg/s (0.0061 rad/s). The red line at 0.35 deg/s indicates a swing-phase detection, while stance-phase detection is signified when the red line is at 0.0 deg/s. The periodic blue line represents raw angular-rate data.

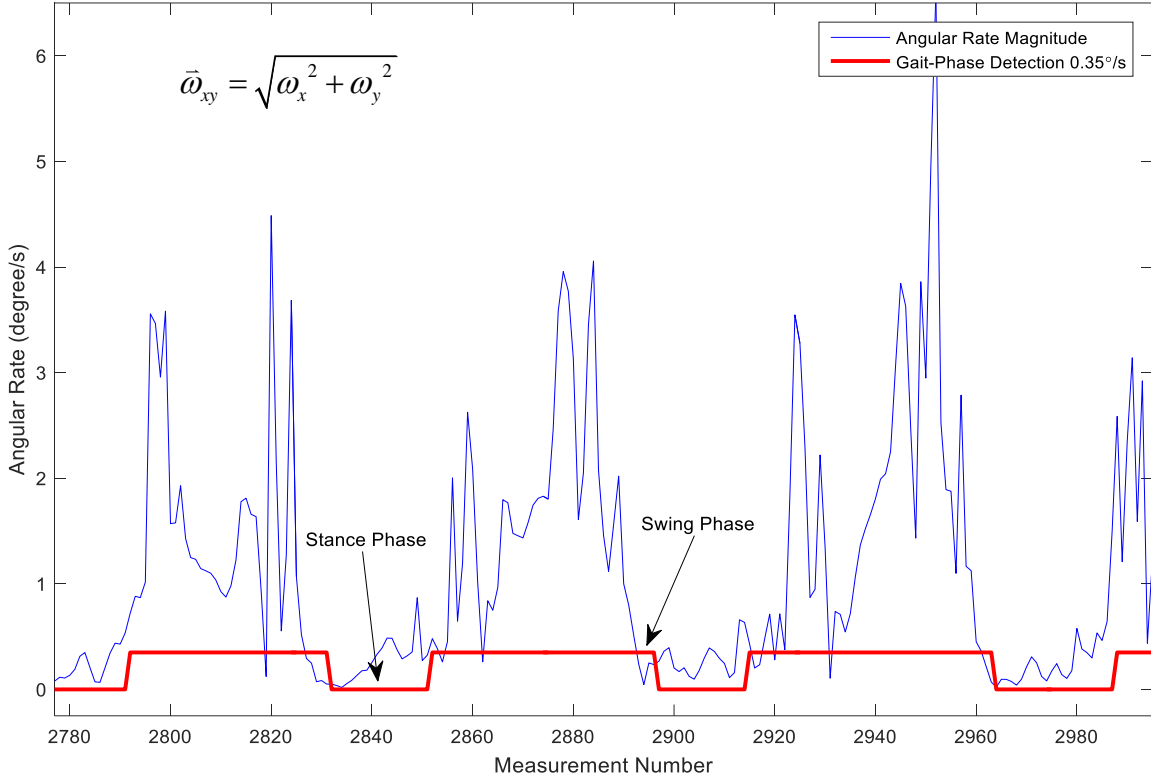


Figure 17. Combined Angular Rate, Sensor 3, Lap 9, Three Footsteps of Gait-Phase Detection, Data Not Squared, SNR Low

The data from three right footsteps are presented in Figure 17. The blue angular-rate line during the swing phase is mostly 1–5 deg/s, and during the stance phase there is an undesirable, non-zero amount of movement picked up by the gyroscopes. Though the right foot of the user was in stance phase, the assumed angular rate of zero was not observed in practice. Instead, a pseudo-error bias was present, owing to the continuous rotation of the foot during walking that added to actual sensor bias as well as a form of pseudo-noise error derived from the minute movements in a planted foot coupled with real sensor noise. The PNS is able to acquire good position accuracy by listening to specific sensor types in a gait phase. The FQA determines roll, pitch, and heading using the accelerometer and magnetometer in the stance phase, while the dynamic rate quaternion sources data from gyroscopes in the swing phase. The data in Figure 17 show a less than desirable difference between the useful swing- and stance-phase measurements. Since the PNS assumes the stance-phase angular rate

quantities as zero, any non-zero output is viewed as noise. With a signal averaging near 2.0 deg/s in the swing phase and 0.5 deg/s in the stance phase, the SNR was much lower than desired, at  $\text{SNR}=10\log_{10}(2/0.5)=6$  dB. This made it difficult to find a suitable angular-rate threshold value, resulting in sub-optimal gait-phase detections.

An improved gait-phase detection method was sought to increase PNS performance. The original angular-rate threshold calculation used for gait-phase transition logic was

$$\bar{\omega}_{xy} = \sqrt{\omega_x^2 + \omega_y^2}, \quad (3.13)$$

where  $\omega_x$  is the gyroscope's X-direction body-frame measurement and  $\omega_y$  is the Y-direction measurement [11]. This formula creates a two-dimensional (2D) vector and was originally chosen in [11] because the inclusion of Z-direction gyroscope data was found to slightly degrade performance. The 2D vector relies on the IMU being mounted such that the gyroscope's measurement plane is aligned to the foot-swinging plane, where the most angular movement occurs. Another shortcoming of the 2D formula is the output does nothing to reduce noisy measurements in the stance phase or increase the signal in the swing phase. This research improves the original formula by making it exponential and incorporating the third Z-axis gyroscope body-frame measurement  $\omega_z$ , adding greater signal strength and reducing noise. This updated version is used in gait-phase detection only. The exponential formula is

$$\bar{\omega}_{xyz} = \omega_x^2 + \omega_y^2 + \omega_z^2 \quad (3.14)$$

where the exponential formula was tested with the same data used in Figure 17, the results used in Figure 18, which illustrates a dramatic improvement, creating a sharp boundary between the swing and stance phases made possible by the exponential formula amplifying the signal while attenuating the noise. Equation (3.14) is effective only because the combined gyroscope output during the swing phase was typically greater than a value of one; therefore, the signal increased with the squared term. Conversely, since the stance-phase noise measurements were mostly less than one, noise was significantly attenuated by the squared term. The result was a new SNR of about 30 dB,

compared to the previous SNR of 6 dB, where Equation (3.13) is used. The higher SNR allows for more robust gait-phase detection by reducing the PNS's sensitivity to the user-selected threshold value. The added stability enables a user to find an optimal gait-phase detection setting easily, ensuring best possible PNS performance.

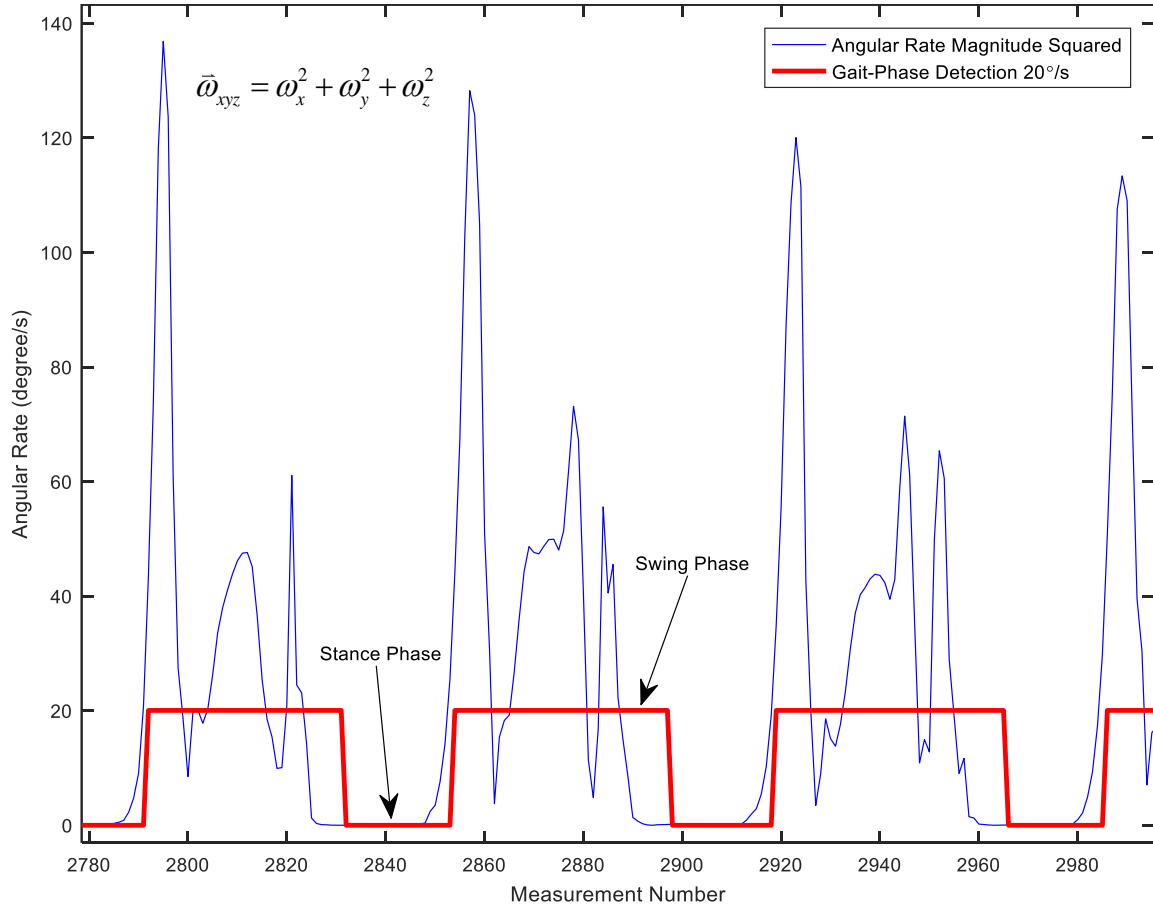


Figure 18. Squared Combined Angular Rate, Sensor 3, Lap 9, Three Footsteps of Gait-Phase Detection, Data Squared, SNR High

Initially, it was hypothesized that the best complementary-filter performance would be achieved when the gait-phase logic immediately picked up a phase change from stance to swing or swing to stance. In practice, this is mostly true but with a few conditions. The gait-phase detection logic has a user setting that controls how quickly the gait-phase changes when faced with measurement data breaking the threshold. The logic was arranged to declare a change in gait phase if a consecutive number of measurements

fell above or below the threshold setting. False alarms were found much more damaging to performance than slow detections. The exponential formula (3.14) has the advantage of creating precise boundaries between phases, as demonstrated in Figure 18. In the swing phase, the angular-rate magnitude very quickly drops below the selected threshold as the walker transitions to stance phase. The original (3.13) results in a slower transition between phases, which makes it harder to select an optimal angular-rate threshold. The PNS performance is also very sensitive to the selected threshold value. A small change in this value can result in an extreme difference in results. In contrast, the exponential-formula threshold value could be set within a wide range of values, such as 6–30 deg/s (0.1–0.5 rad/s), and show very little difference in the returned PNS position. This characteristic adds much needed robustness to the PNS.

### C. TUNING

Optimal performance of the complementary filter was achieved by tuning it using real-life data. To do so, various parameters were altered, with the gain parameter  $k$  being most important. Adaptive gain was chosen over constant gain because it is demonstrated in [11] to offer superior performance in the PNS.

Within the PNS, and subsequently the complementary filter, the FQA and dynamic rate quaternions run simultaneously. Since the complementary filter is adaptive, there are effectively two gains,  $k_s$  and  $k_d$ , which are mutually exclusive and used in the stance and swing phases respectively. The gain dynamically changes the FQA's weighting based on the current angular-rate measurement's relationship to a selected threshold value. The concurrent nature of these static and dynamic quaternion operations allow a blended-filter approach, enabling smoother transitions between sensors.

#### 1. Athletic Track: Walking Only

The complementary filter was tuned using data collected from the second test event. In this test, the user circled an athletic track with a 3-Space module mounted on the right foot. This test did not incorporate running or alternative postures (e.g., kneeling or prone). Insights from a preliminary investigation to determine optimal sensor settings,

equipment, and test setup were implemented, such as fashioning a rigid mounting bracket and calibrating sensors in the field.

*a. Test Setup*

To limit extraneous influences and simplify tuning, variables were tightly controlled. A rubber-surfaced athletic track was selected as the test location, as shown in Figure 19. A measuring wheel was used to find the true circumference of one complete lap around the track. The measuring wheel was rolled along the white line that divides the inside lane from the next lane over, resulting in a measurement of 404.6 m. Both straight legs were 100.0 m, while each semicircular turn added a little over 102.0 m to the distance. All walking tests occurred at the same track, the user followed the same white line in the same direction using identical start and finish points.

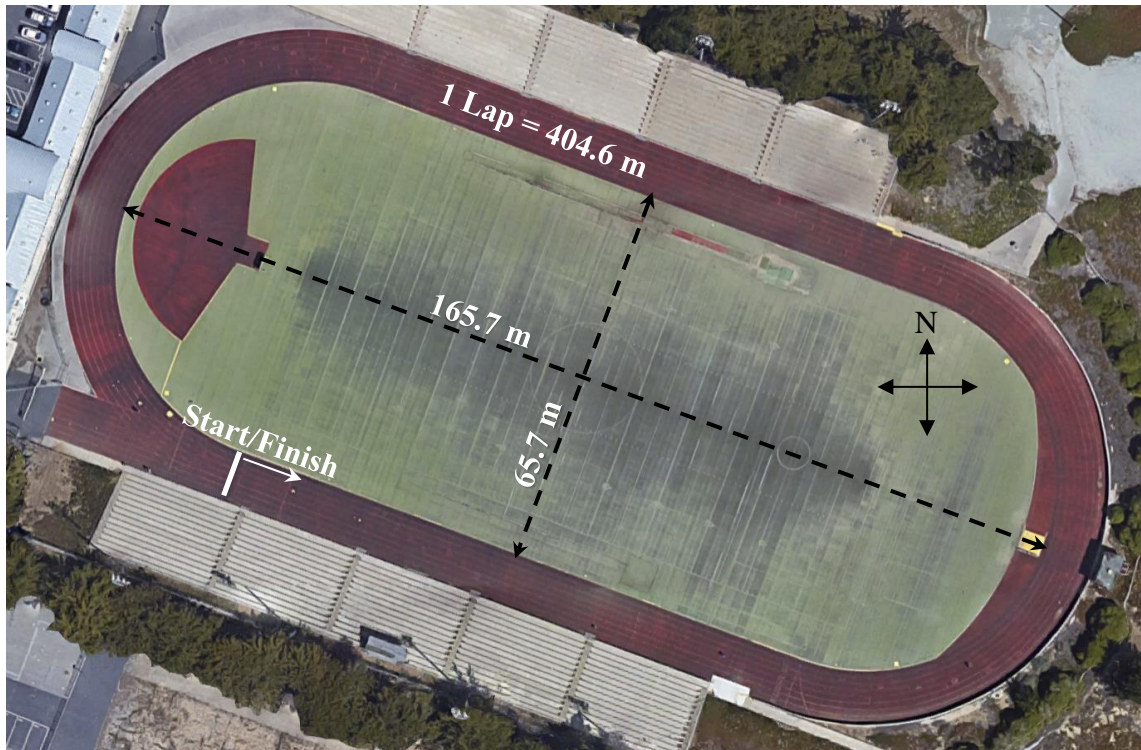


Figure 19. Athletic Track Used for Testing. Adapted from [64].

Considerable thought was given to the mounting of the 3-Space module. The preliminary investigation used stretchy rubber brackets supplied by the manufacturer, which attached to an elastic belt by means of Velcro straps, permitting undesirable sensor movement during walking. While later analysis provided reassuring evidence that the PNS was largely unaffected to adverse conditions such as jostling, it was an obstruction in tuning of the complementary filter and characterizing a performance baseline. Further testing addressed these concerns by designing and manufacturing custom mounting brackets.

***b. The PNS Bracket***

A stable, rigid bracket was created for testing purposes, using computer-aided design to 3D print a rigid plastic bracket in  $2 \times 1$  and  $2 \times 2$  versions, which carry a maximum of two or four IMUs respectively. These can be seen in Figures 20–23. In this paper, the naming convention of brackets are defined as the leading number “2” corresponding to the number of columns of IMUs, and the trailing number “1” or “2” corresponding to the number of rows of IMUs. The rapid prototyping capabilities offered by a 3D printer proved to be beneficial, allowing design evolution to occur within a few hours. Besides adding stability, the brackets also simplified the co-location of multiple IMUs.

For testing the mounting of multiple 3-Space modules at the same location, we predicted that maintaining a constant orientation among the units would simplify their data averaging into a fused output. We also assumed that the IMUs would more likely experience near identical forces if they shared the same rigid bracket. Finally, mounting multiple co-located sensors in one bracket was predicted to assist in a rigorous comparison of performance discrepancies among alternative sensor combinations. The  $2 \times 1$  brackets are displayed in Figures 20 and 21.



Figure 20. 2×1 Sensor Brackets, Left-Foot Bracket (Left) and Right-Foot Bracket (Right)

Brackets were laced to the shoe at the lower instep, just above the intersection of the arch and ball of the foot, as displayed in Figure 21. While the modules may appear to pop out easily, they are tightly secured by the bracket's sidewalls. Only one sensor module came loose in 20 laps of testing.



Figure 21. Right-Foot 2×1 Bracket Holding Two IMUs

The first test setup incorporated two 2×1 brackets, each holding two 3-Space modules as shown in Figure 20. For all foot-based testing, sensor modules 1 and 2 were

mounted on the left foot and modules 3 and 4 on the right. This was done to enable proper comparison between test runs, simplify comparison of the performance differences among individual sensors on the same foot, and ascertain the benefits of averaging co-located sensors.

The second test setup used a 2×2 bracket to co-locate four sensors, as shown in Figure 22, allowing a near-identical force environment in which sensors were compared directly and two to four sensors were fused into a single averaged output.

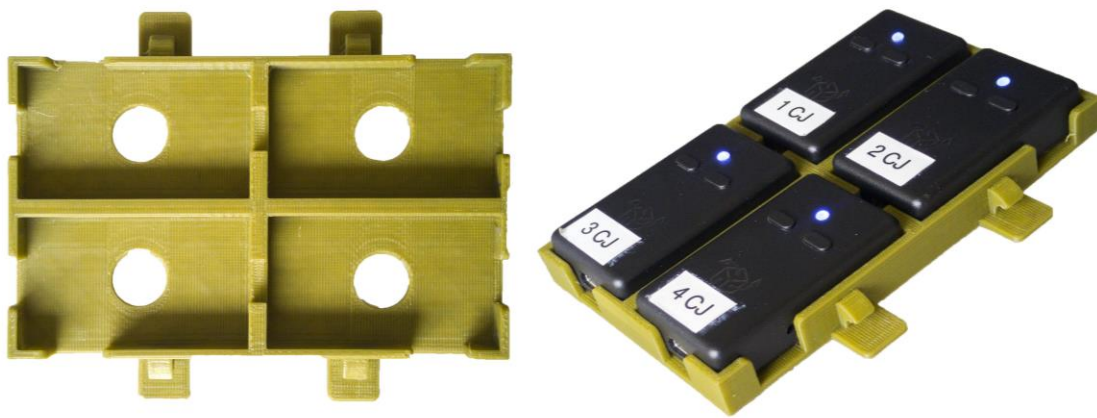


Figure 22. 2×2 Sensor Bracket Holding Four IMUs in Fixed Relationship

The 2×2 bracket was placed on the right foot and, like the 2×1 version, secured by shoelaces as shown in Figure 23. Throughout all test events, only one test run was aborted because the bracket came unhooked. No damage was observed to any component.

Despite being rigid, there may have been some flexure in this and any of the other brackets. This may have contributed to each IMU feeling a slightly different force and subsequently needing slightly different PNS settings, which will be discussed later, to achieve best performance. In addition, each IMU was in a different physical location and could have had small discrepancies in the experienced forces. More than likely, these and several other small contributing factors cause discrepancies between individual IMUs as well as their mounted locations.



Figure 23. 2×2 Bracket Mounted with Four IMUs on Right Foot

*c. Consistent Movement*

Great care was taken to maintain uniform test conditions and execution among test runs. The user carried a tally counter to track right footsteps and for consistent cadence, walked in synchronization with a smartphone metronome application, such that at every beep, the right heel struck the ground. To set walking speed, a comfortable pace was found and the metronome beat adjusted to match. This resulted in a setting of 55 beats per minute (BPM), which constitutes a step frequency of 1.83 Hz when considering both feet. It was noted that [11] used a step frequency of about 1.65 Hz (49.5 BPM), but it was considered unlikely that detectable performance differences could be attributed to such a small discrepancy. Step frequency is the combined total number of steps taken within a given period by both feet, not just the right foot. A normal step frequency for the tester's age group is 2.0 Hz [65]. The difference in step frequency may reflect the tester's above-average height or the attempts of other researchers to control their walking.

*d. Calibration*

The first three walked laps used a pair of 2×1 brackets containing sensor modules 1 and 2 on the left foot and modules 3 and 4 on the right. Before the start of the first lap, a

complete field calibration was performed on each IMU using the sphere-calibration wizard from the 3-Space sensor suite to calibrate the accelerometer and magnetometer. This was accomplished by connecting the 3-Space sensor module to a laptop computer through a USB to micro-USB cable and running the program. Then, the gyroscope was calibrated by using a separate software function within the suite. Previous familiarization trials had revealed the 3-Space sensor module's sensitivity to magnetic fields created by nearby electronics. When a laptop computer was within 0.5 m of a 3-Space module, the magnetic field emanating from the computer's circuitry was powerful enough to distort sensor calibration, resulting in poor PNS performance. Care was taken to maintain at least 1.5 m between the sensor and computer running the calibration software, a distance that eliminated interference from the laptop's magnetic field. Calibration of the accelerometers and magnetometers were considered complete when the estimated density level reached 12, the same value attained previously with PSD characterizations. After the accelerometers and magnetometers were calibrated, the 3-Space module was gently placed flat on a concrete surface for gyroscope calibration. Ferrous material in the concrete, such as rebar or other magnetic interference, was not expected to affect gyroscope calibration because gyroscopes are not nearly as sensitive to magnetic fields as magnetometers are. Afterward, all settings were committed and saved to the 3-Space sensor module's non-volatile memory. These calibration steps were repeated for all four sensors, taking about 15 minutes total.

*e. Track Testing*

The first round of testing consisted of six individual laps, each with two different sensor configurations. Laps 1–3 tested the 2×1 brackets, while Laps 4–6 tested the 2×2 bracket configuration. The IMUs were set to begin recording data with the press of a physical button on the 3-Space module and stop with the press of the other button. Each start-and-stop interval created a separate file, so that each lap's measurements were recorded in a discrete TXT data file on the sensor module. These files were later used for post-processing and data analysis.

Before the fourth lap, all four modules received a quick gyroscope calibration and were placed on the right foot in the 2×2 bracket, as demonstrated in Figure 23. This configuration is in contrast to Lap 1 where two sensor modules were mounted on each foot using the 2×1 brackets. Three laps with the 2×2 configuration were then completed. Before the sixth and final lap of the first evening’s testing, the sensors were adjusted so that a USB cable could be attached to the end of the IMUs while they were already inserted in the bracket to allow for recalibration of the gyroscopes. This new arrangement was accomplished by rotating the topmost sensors, 1 and 2, by 180° so that their micro-USB ports were unobstructed. This measure was taken on the hypothesis that jerk forces encountered when snapping the sensor modules into the bracket might corrupt a previous gyroscope alignment. Later analysis proved this to be an overly cautious precaution; the sensors were much more resilient than expected.

## 2. PNS Settings Explained

Collected sensor data were processed through the PNS algorithm using default values originally optimized in [11] for the MicroStrain 3DM-GX1, as shown in Table 2.

Table 2. PNS Settings, Optimized for MicroStrain 3DM-GX1. Source: [11].

Mounting Location: Right Foot	Value
Gait-Phase Detection Angular-Rate Threshold (deg/s)	0.35
Samples Above/Below Threshold Required to Change Gait Phase	5
Samples to Save	20
Complementary Filter Angular-Rate Threshold (deg/s)	0.35
$k_s$	1.0
$k_d$	0.00075

The first value given in Table 2 is the gait-phase detection angular-rate threshold value, measured in degrees per second. This threshold was used by the gait-phase-detection algorithm to determine whether the user was in a stance phase (if below) or swing phase (if above).

The second parameter signifies the consecutive samples above or below the threshold required to change the gait phase. This value adds a hysteresis used to prevent momentary, false gait-phase changes caused by large-magnitude angular-rate error spikes. False gait-phase detections degrade PNS performance by incorrectly weighting each sensor type. An example illustrating how this value increases the robustness of the system is presented in Figure 18. Note that there appears to be a delayed reaction in the system's detection of a gait-phase change. If the setting is set to a value of 5.0, the gait-phase detection logic requires five consecutive samples of data above or below the angular-rate threshold before it agrees to change the gait phase; a number of factors are used to determine the best value for this parameter. Setting an angular-rate threshold too small (e.g., 0.1 deg/s) was shown to result in false detections; the logic mistakenly registered a swing phase though the tester was in stance phase. Conversely, a problem can occur when setting the threshold too large, the algorithm does not detect a stance phase event. Accuracy is degraded in each of these scenarios because the gain values change for each sensor type based on gait phase. Having a threshold value that triggers false alarms or misses events reduces the accuracy of the PNS because the wrong gain value is processed with the measured data.

The third parameter value in Table 2 is the "Samples to Save Setting." This parameter acts as a buffer; its value determines the number of previously reported velocity measurements to use.

The fourth parameter value signifies the complementary filter's angular-rate threshold, which is similar to the gait-phase detection threshold. The difference is that this value determines the adaptive-gain boundary threshold. In testing, improved performance was achieved when able to change both angular rate thresholds independently as needed to boost PNS performance.

The gain values  $k_s$  and  $k_d$  are, respectively, the static and dynamic gains. Static gain is used when angular-rate measurements falls below the complementary filter's angular-rate threshold setting; this indicates the foot was in the stance phase. Dynamic gain is used only when angular-rate output is above the threshold setting, which indicates the foot was in the swing phase.

As stated earlier, the purpose of the two gains is to incorporate adaptive weighting with the FQA. Conditional weighting is used to detect attitude and heading from accelerometer and magnetometer measurements. If the gain shows a high value such as  $k = 1$ , the accelerometer and magnetometer measurements have equal say with respect to the gyroscope in the complementary filter. If the gain is set very low, such as  $k = 0.01$ , then the weighting of the accelerometer and magnetometer are significantly attenuated compared to the gyroscope. This weighting reduction allows the gyroscope to become the dominant source of information for the complementary filter.

### **3. PNS Performance when Walking**

A set of 3-Space foot-mounted measurement data was processed through the PNS using gain settings found to work best with the MicroStrain 3DM-GX1 [11]. These post-processed data were then plotted in Figure 24, displaying suboptimal results. The trial began at point (0, 0), and the user walked the same counterclockwise path for every run. Each blue dot represents the right foot's placement when a stance phase is detected. Note that the start and end points do not match, though the tester ended the lap in the same location as starting. The step-plot demonstrates the need to custom tune each IMU, an undesirable but not surprising requirement. The main problem illustrated by Figure 24's step-plot is the PNS's inability to accurately detect turns using the original settings. The end result was a large misalignment error of 22.63 m between the starting and ending points. Custom tuned PNS settings are required to achieve adequate performance.

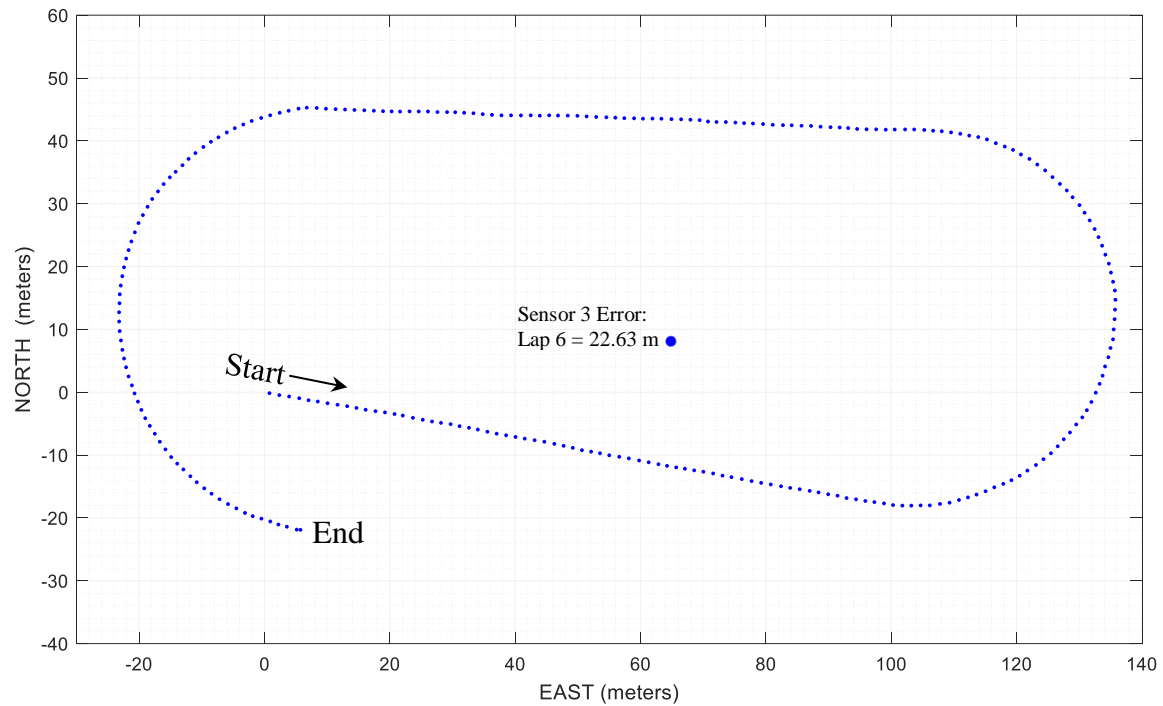


Figure 24. Step-Plot of 3-Space Sensor Using MicroStrain 3DM-GX1 PNS Settings; Sensor 3, Lap 6

After many attempts to improve step-plot performance by adjusting each setting, optimal values for Sensor 3's sixth lap were found. These are provided in Table 3.

Table 3. PNS Algorithm Values, Optimized for 3-Space Sensor

Mounting Location: Right Foot	Value
Gait-Phase Detection Angular-Rate Threshold (deg/s)	0.35
Samples Above/Below Threshold Required to Change Gait Phase	3
Samples to Save	15
Complementary Filter Angular-Rate Threshold (deg/s)	0.101
$k_s$	1.0
$k_d$	0.01175

With these new settings, the PNS was able to process Sensor 3's lap-six measurements and achieve a much-improved outcome, as shown in Figure 25. This time, the PNS performed well at detecting turns, thereby reducing the alignment error of the traversed path. Despite improved performance, there remained an error of 5.41 m. The resulting step-plot suggests the user reached the original starting location, but proceeded another 5.41 m. In reality, he stopped precisely at the starting location. The reason for this overlap error is assumed to be inaccuracies in the sensors themselves. Despite this modest error, the gait-phase detection algorithm properly detected all 266 steps made by the right foot. The total distance traveled is shown as 396.0 m, not the actual distance of 404.6 m that was previously measured.

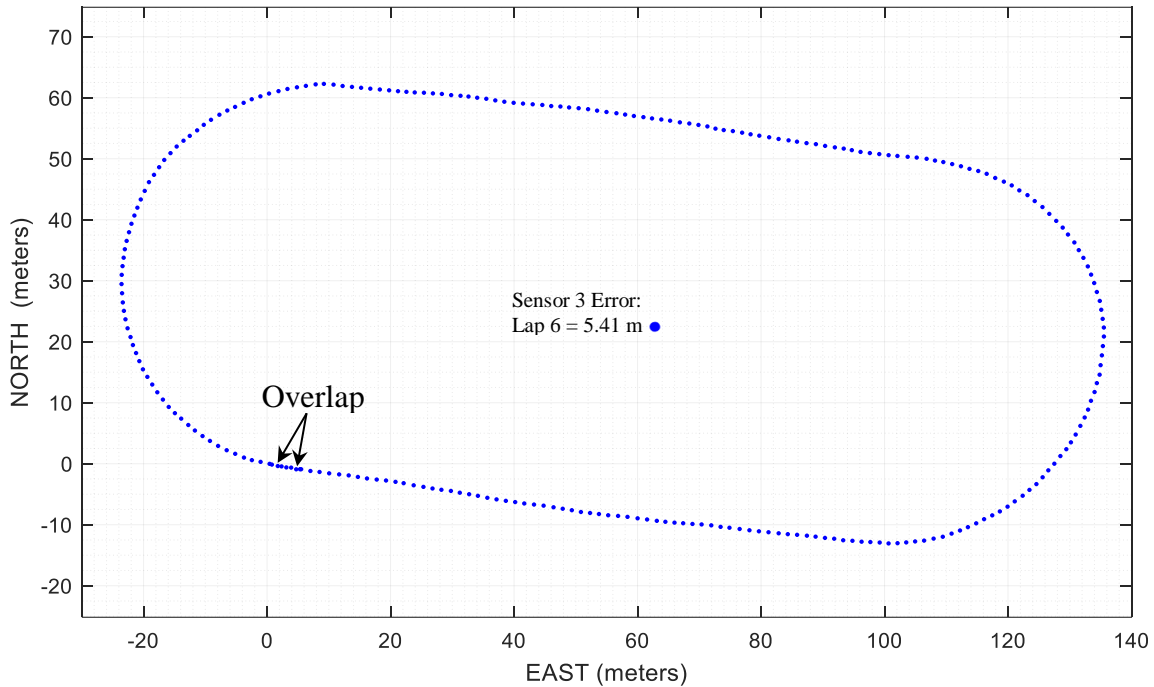


Figure 25. Step-Plot of 3-Space Sensor using Optimized PNS Settings; Sensor 3, Lap 6

The total-distance-traveled error is described as a scaling error, hypothetically attributable to the mounting of the IMUs at the top of the foot, not the bottom. The leg rotates about the ball-and-socket joint at the hip. The velocity at each point increases as distance from the hip grows. The result of this increased moment arm is that the forces

detected during the swing phase by the IMU will be smaller in magnitude above the foot than below. The lesser-magnitude velocity from the top of the foot was integrated, resulting in a shorter distance than truth.

*a. Multiple Laps Using Same IMU*

One problem encountered was performance inconsistency between laps. The complementary filter may be optimally tuned for a specific sensor and test lap, but the same sensor under nearly identical conditions tends to require retuning. An example of the need for custom tuning is presented in Figure 26, which only displays Sensor 3 PNS outputs but uses tuning values optimized for Lap 6, on Laps 4 and 5 as well. Note the significant variation among traversed paths, as well as the general noise of the data. These variations occurred despite efforts to keep as many variables as possible constant: same path, temperature, step frequency, 2×2 mounting bracket, etc. One possibility for the path variations may be that the gyroscope was recalibrated just before Laps 4 and 6. Another potential reason is that commercially available IMUs offer limited performance.

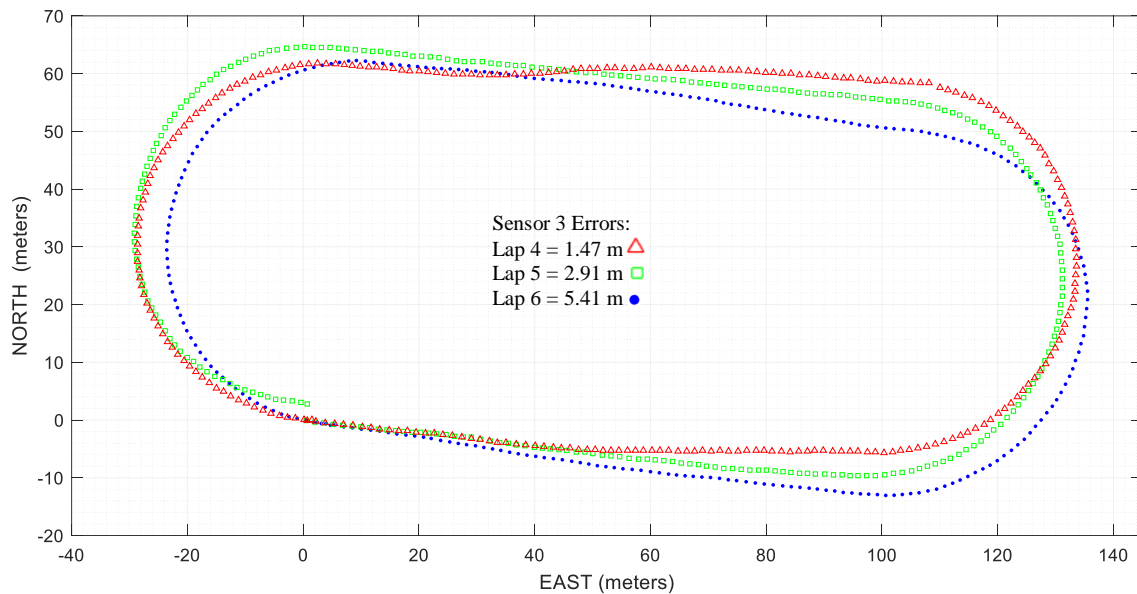


Figure 26. Sensor 3's Step-Plots for Laps 4–6 using 2×2 Bracket and Settings Optimized for Lap 6

Using a gyroscope as an example of sensor quality, we find they all have several dominant sources of error, including gyroscope bias instability, scale-factor non-linearity, and angular random walk. These errors combine to contribute to inconsistent and inaccurate position solutions. Tuning the PNS greatly reduces these negative characteristics, but the step-plots presented in this thesis demonstrate that the PNS can only do so much. Even with strict control of variables, where all laps are traversed at exactly the same pace and along the same path, the resultant step-plots still displayed variations.

Further analysis indicated that small changes in how each IMU was mounted to the foot could cause large performance differences; these differences can be seen in in Figure 27. A small change in the physical environment, such as placing Sensor 3 in a 2×1 bracket, seemed to make a large difference in estimated starting and stopping locations and overall path.

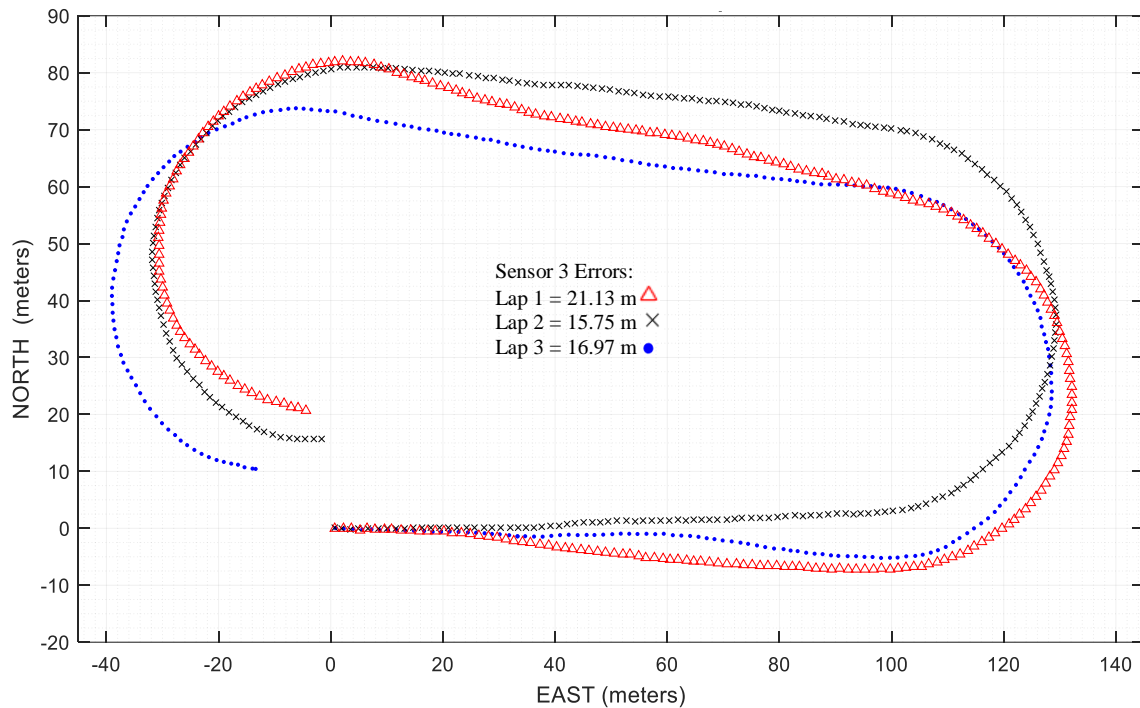


Figure 27. Sensor 3's Step-Plots for Laps 1–3 Using 2×1 Bracket and Settings Optimized for Lap 6; Gains Not Tuned for Each Lap

In Figure 27, Sensor 3's gyroscope only received an initial calibration, it was not recalibrated between laps, and performance for Laps 1–3 is noticeably degraded compared to Laps 4–6 in Figure 26. The takeaway from these results is that seemingly insignificant variations in the test setup may create large variation in PNS performance.

***b. Multiple IMUs from the Same Lap***

To compare multiple IMUs in the same lap, all four sensor modules from Lap 6 were analyzed. Since they experienced near-identical conditions, with all modules mounted on the same 2×2 bracket, their performance was expected to be very close. The complementary filter settings for Sensor 3 were optimized for the sixth lap, and the other sensor modules were processed with those same settings. Afterward, the sensors were processed through the PNS and plotted as shown in Figure 28.

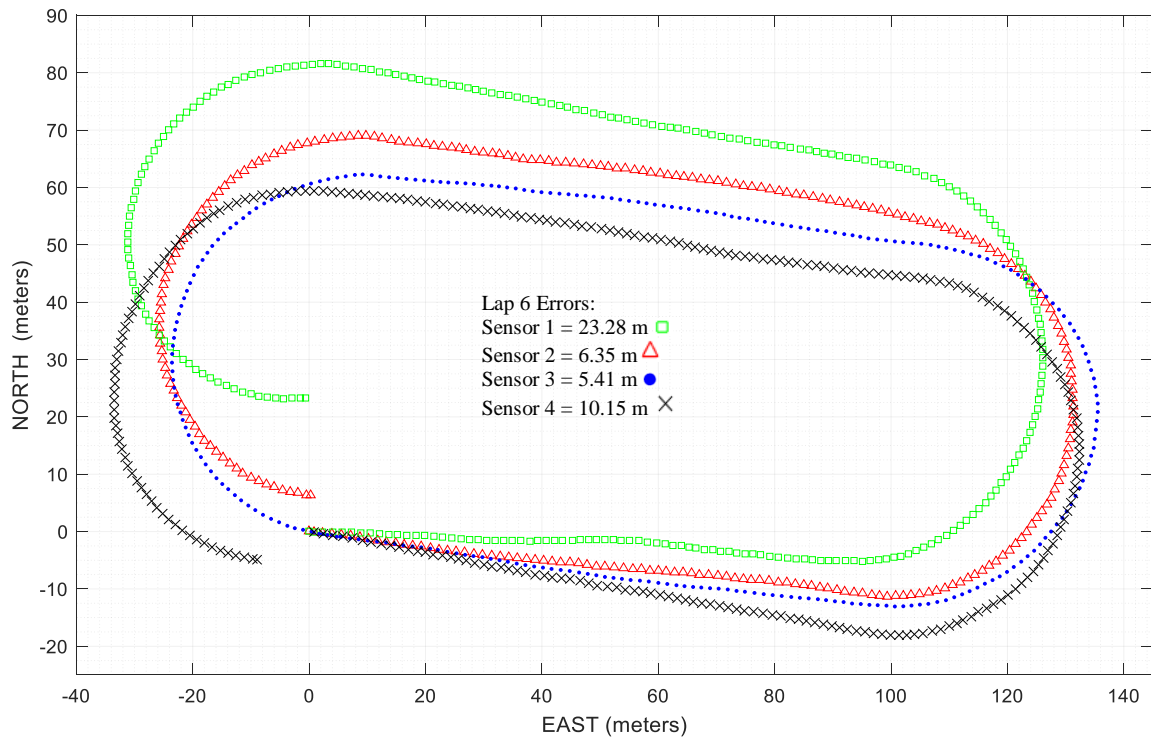


Figure 28. All Four Sensors Step-Plots for Lap 6 Using 2×2 Bracket and Settings Optimized for Sensor 3's Lap 6; Gains Not Tuned for Each Sensor

Interestingly, when compared with the previous section, there were fewer performance variations from one IMU on multiple test runs than from multiple IMUs on a single test run. These results amplify the point that the best way to reduce PNS error is to tune the complementary filter for each IMU, and then for each test run. Since this is not practical in any reasonable real-time implementation of the PNS, the next alternative is to fine tune the complementary filter for each specific sensor—Sensor 1 with different settings from Sensor 2, and so forth. Unfortunately, until low-cost IMU technology is considerably improved, extra care is required to achieve acceptable performance in a real-time PNS by ensuring each sensor module is properly tuned before operation.

The PNS provides a reasonably good position solution when used in conjunction with a low-cost IMU but only after custom complementary filter tuning has been applied. PNS performance is subject to substantial variations among IMUs and the conditions in which they operate. It is cumbersome to retune the PNS for every test event and likely impossible in real-time. It is, therefore, concluded that a single, low-cost, currently available, consumer-grade AHRS IMU cannot provide the measurement quality needed for acceptable position accuracy.

### *c. Temperature Sensitivity*

To ensure robust data collection, additional laps were performed a week after those in the previous section. Like the others, these were conducted in the evening, but this time the ambient temperature was 5°C (9°F) lower at 13°C (55°F). The IMUs were exposed to the open air longer than before, allowing more acclimation to ambient conditions before calibration was performed.

For unknown reasons, IMU performance was significantly better than in previous testing. A possible explanation is the lower ambient temperature and a longer delay prior to calibration. Temperature correlates with sensor noise; therefore, a lower temperature may result in reduced sensor noise (i.e., lower ARW). In addition, exposing the IMUs to the ambient temperature for ~20 minutes before calibration might have reduced the total impact of non-linear errors caused by temperature changes after calibration.

The step-plot in Figure 29 shows all three laps of measurements taken from Sensor 3, encased in the 2×1 bracket and mounted to the right foot. Complementary-filter tuning revealed that for Sensor 3 to enable the PNS to create step-plots in the shape of the athletic track, the static gain  $k_s$  must be set to a value of 1.0 and the dynamic gain  $k_d$  should be 0.091.

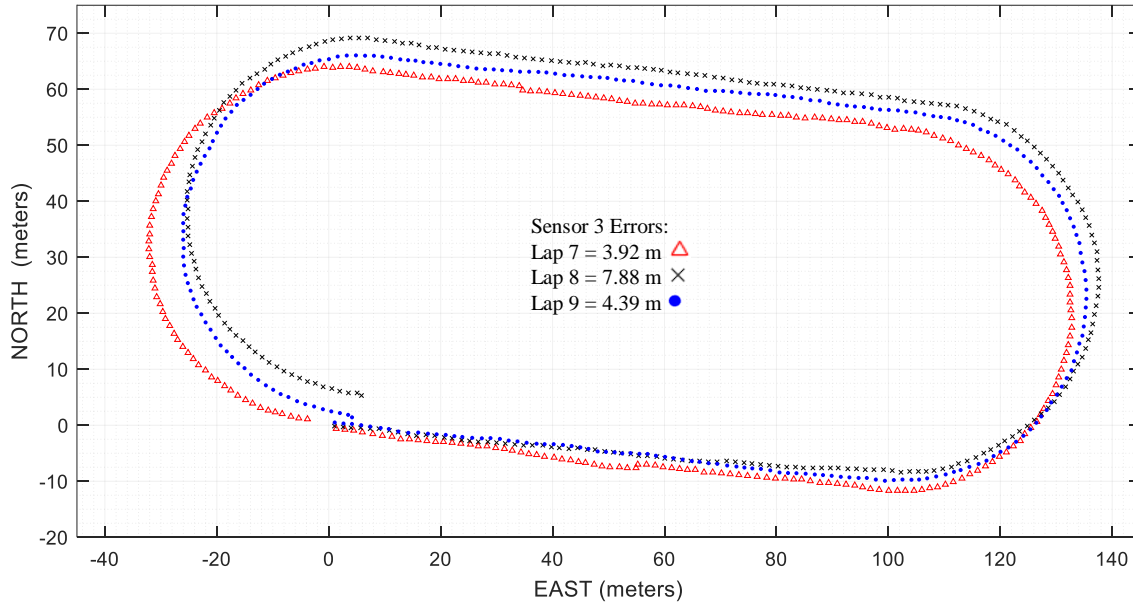


Figure 29. Sensor 3's Step-Plots for Laps 7–9; Gains Tuned for Each Lap

To further demonstrate the performance-altering effects of custom complementary filter tuning, gains that had been optimized for Sensor 4 were used to process Sensor 3's data. The corresponding degraded PNS step-plot is presented in Figure 30. At first glance, error values may appear to be very close to the optimally tuned step-plot in Figure 29, but data used to generate Figure 30 produced a plot with an elongated track and a rougher overall path.

Despite being an identical IMU compared to Sensor 3, Sensor 4 still had performance differences. Perhaps, even though effort was made to maintain consistent calibration methodology between each IMU, small differences were large enough to

change overall performance. Another possibility is the fact that each sensor within an IMU does not have the exact same performance as demonstrated by their PSD.

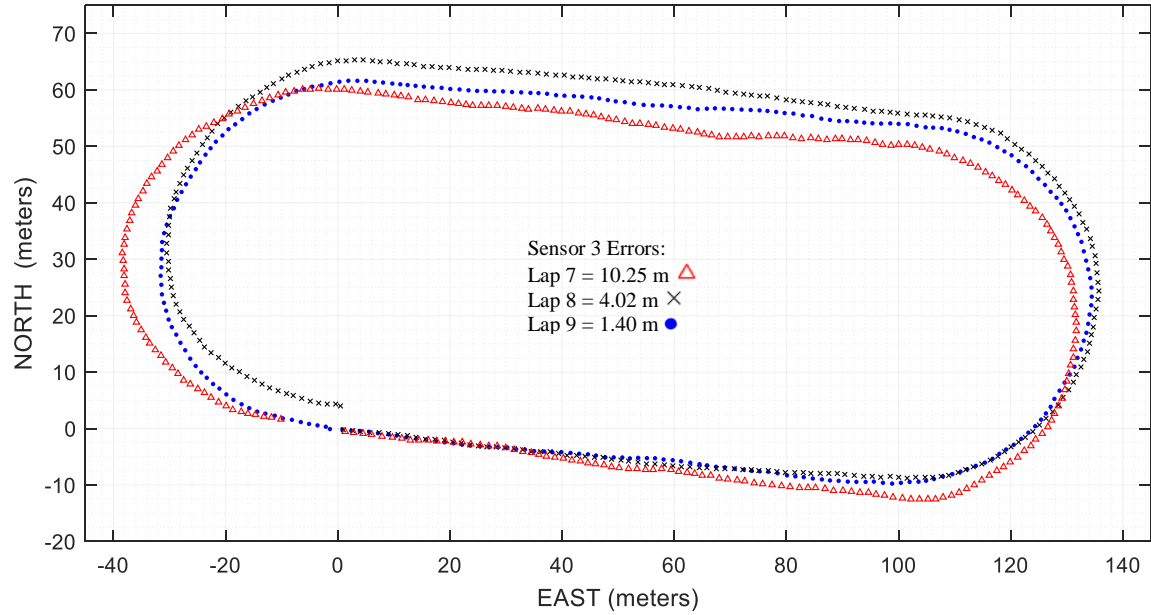


Figure 30. Sensor 3's Step-Plots for Laps 7–9; Gains Tuned for Sensor 4's, Lap 2

The PNS settings used to process Sensors 3 and 4 are displayed in Table 4. Note that the gait-phase detection threshold and complementary-filter thresholds are the same. This is because sensor performance from that round of testing was more robust. The values of 0.35 deg/s were found optimal, but a wide span of values from 0.1–0.5 deg/s worked nearly as well.

Table 4. PNS Settings Optimized for Sensors 3 and 4 in Laps 7–9

Mounting Location: Right Foot	Value Sensor 3	Value Sensor 4
Gait-Phase Detection Angular-Rate Threshold (deg/s)	0.35	0.35
Samples Above/Below Threshold Required to Change Gait Phase	5	5
Samples to Save	20	20
Complementary Filter Angular-Rate Threshold (deg/s)	0.35	0.35
$k_s$	1.0	1.0
$k_d$	0.091	0.01175

Sensor 4 was found to perform better than Sensor 3, as evident by the more accurate step-plot data displayed in Figure 31. Sensor 4 was tuned to work best in the eighth lap as illustrated by the small error value of 1.82 m, while the two other laps did almost as well. This step-plot appears to be smoother for two of the three laps and more accurate than the data extracted from previous testing.

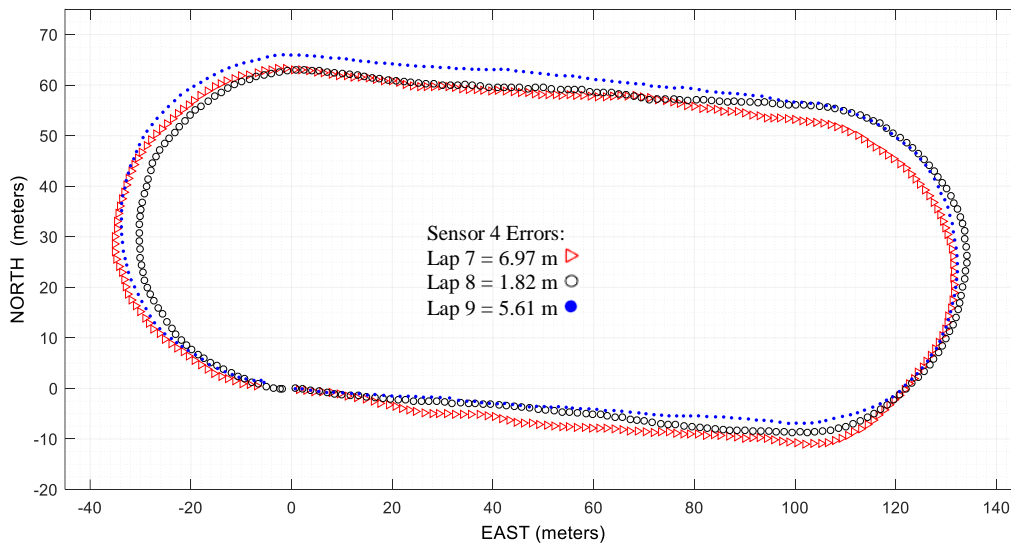


Figure 31. Sensor 4's Step-Plots for Laps 7–9; Calibration Occurred at Lower Temperature; Settings Optimized for Lap 8

#### **D. PLACEMENT**

The results of placing IMUs at different locations on the body are investigated in this section. Each body location studied had its corresponding IMU and PNS performance analyzed. A total of eight 3-Space modules were strapped to a single user: two on each foot, one on each shin, one on the lower-back, and one on the chest. The manufacturer-provided mounting straps were elastic and the sensor mounts were blue, stretchy rubber. The distribution of sensors on the user is displayed in Figure 32.

Each mounting location tested was selected according to a specific rationale. The foot sensors were used as a performance baseline because their data can be used with the complementary filter with optimal gain settings as discussed previously. Two IMUs were placed on each foot to provide a feel for the variation in sensor performance when mounting location is held constant. A module was mounted on each shin to evaluate its potential use with [10]’s posture-finding algorithm. The lower back was tested under the hypothesis that an IMU will see better performance when insulated from acceleration and jerk forces, like those experienced with the foot. The lower back was chosen after correspondence with a biomechanics expert who confirmed the small of the back as the most stable location, with the least movement [66]. The chest was chosen primarily to allow [10]’s posture-finding algorithm to work, though it was predicted that PNS performance would be suboptimal. The bending and turning of the chest independently of walking direction was expected to degrade PNS accuracy. The head was briefly considered but ruled out as an acceptable location because people turn their heads frequently, and the user would potentially have to wear a cumbersome bracket, maybe even a helmet, to operate the PNS.



Figure 32. Distributed IMU Mounting Arrangement; Two per Foot, One per Shin, One on Lower Back, and One on Chest, for Eight Total

Placing an IMU on the foot was originally motivated by an educated guess, which predicted that location would be optimal for gait-phase detection since it experienced the largest forces, owing to its location at the extremity of the body's longest moment arm, which rotates about the hip. A long moment arm produces a large angular-rate SNR to allow proper gait phase detection between the swing and stance phases, enabling the complementary filter to quickly and accurately identify footsteps and integrate them into position estimates. For this thesis, “signal” corresponds to the amplitude of the angular velocity experienced by the 3-Space module during the swing phase, and the “noise” is the angular velocity experienced during the stance phase. These usages depend on the presumption that there is no angular rate during the stance phase, which for the purposes of this thesis is assumed to be true.

Another motivation for multiple IMU mounting locations was the concern that a foot-mounted module's performance might be diminished by impulse forces when the heel struck ground; as a result IMU performance might be improved by moving the 3-Space module to a less dynamic place on the body. In this section, the effects of an

IMU's placement on different parts of the body were investigated in order to test the assumptions regarding the different mounting locations.

This particular set of testing did not employ the 3D printed brackets created for this research, and testing took place during the day. Lower control of variables such as weather, brackets, and step cadence was not expected to degrade outcomes because only relative performance of different mounting locations was of interest as opposed to the absolute performance and accuracy of the PNS system.

### **1. The Shin**

The shins are an attractive choice for IMU placement because they are exposed to angular forces like those of the feet but with less jerking and rolling. A smooth but distorted step-plot was created using data collected from a shin-mounted IMU, as shown in Figure 33. In this test, the 3-Space sensor module was mounted to the tester's left shin, held firmly by a stretchy rubber bracket and wrapped in elastic Velcro as shown in Figure 32.

A multiplication factor of 1.094 was used to properly scale up the step-plot, because the PNS calculated the total distance walked as only 370.0 m instead of the actual distance of 404.6 m. The multiplication factor brought the total distance traveled to within 0.5 m of true. Scaling each mounting location to negate path compression was needed to normalize the data and effectively compare performance at different locations.

Compared to the foot, the shin's path-compression error was most likely due to the shortened lever arm from the hip, which reduced perceived forces.

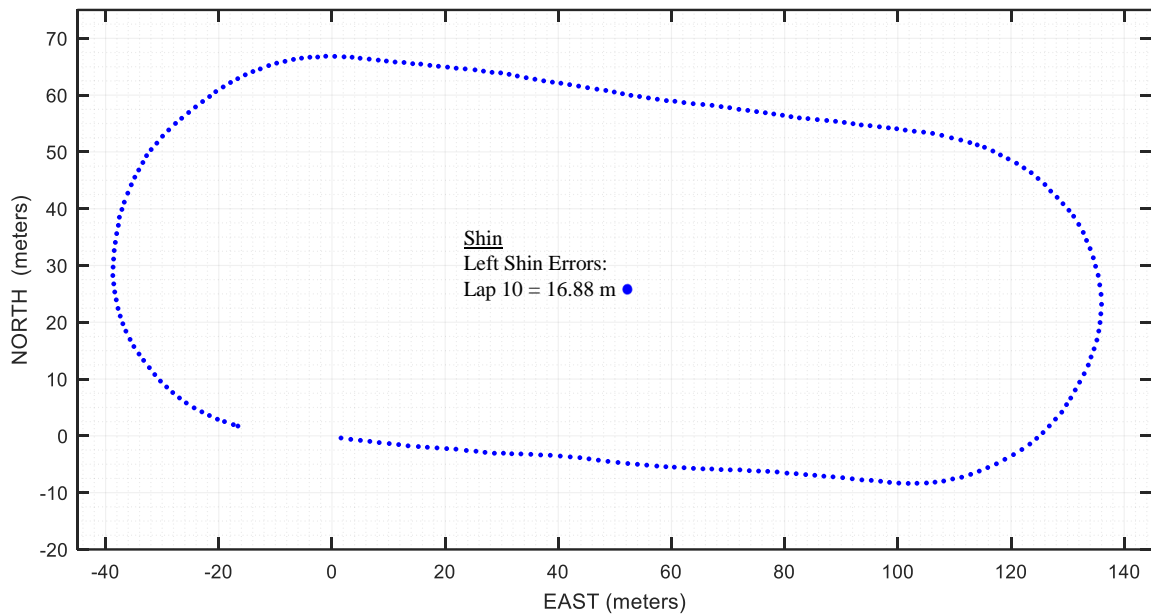


Figure 33. Left-Shin Mounted IMU's Step-Plot

The complementary filter settings for the shin-mounted module were adjusted until optimal parameter values were found. “Optimal” is used to describe settings that achieve a step-plot with characteristics including a smooth path in the shape of an athletic track and a minimal error value. “Error value” represents the difference between the estimated starting and stopping locations, not to be confused with total-distance traveled or scaling (i.e., path-compression) errors. There was no accurate TSPI truth source (e.g., GPS) mounted on the user during testing, so errors could not be calculated with every measurement.

The optimal shin settings are listed in Table 5. The specific 3-Space module used for shin testing was not one of the sensors used in other parts of this research, that is, not Sensors 1–4. The selection of the left shin sensor for analysis was arbitrary—there is no reason to favor left over right.

Table 5. Optimal Settings for Left-Shin Mounted IMU, Lap 10

Mounting Location: Shin	Value
Gait-Phase Detection Angular-Rate Threshold (deg/s)	1.0
Samples Above/Below Threshold Required to Change Gait Phase	5
Samples to Save	20
Complementary Filter Angular-Rate Threshold (deg/s)	1.0
$k_s$	1.0
$k_d$	0.005
Multiplied by Scale Factor	1.094

Two parameters of note are depicted in Table 5. The first is the gait-phase detection angular-rate parameter, which was set to 1.0 deg/s. This is quite a bit higher than the 0.35 deg/s setting, which was optimized for foot-mounted sensor modules. This implies the angular-rate measurements remained high, even in stance phase, which makes sense because, while the foot may be planted on the ground, the shin rotates about the ankle with a lever arm several centimeters above it and a non-zero angular rate of rotation. The second parameter setting of note, is the reduced weighting of the dynamic gain for the FQA, where  $k_d = 0.005$ . This is similar to the gains found to work well with foot-mounted IMUs. This low gain is a sign that the gyroscope was the dominant sensor used for position calculations during the swing phase. The output position values were multiplied by a scale factor to normalize the shin step-plot with the foot-mounted step-plots.

To further illustrate the point of increased noise in a shin-mounted stance phase, an angular-rate length plot is provided in Figure 34. Unlike data derived from foot sensor

modules, shin-mounted angular-rate measurements make it difficult for the gait-phase detection algorithm to operate correctly because of excessive noise near or above a value of one. This prevents the use of the SNR-boosting exponential Equation (3.14), where angular rate values less than one are attenuated and values greater than one are amplified. Instead, Equation (3.13) was found to work best for the shin, performing a square root of the sum-of-squares operation using only X- and Y-axis gyroscope measurements to compute the angular-rate magnitude. The resulting SNR was about 8 dB. Compare that to an SNR of 30.0 dB for a foot mounted IMU accompanied by the exponential formula (3.14).

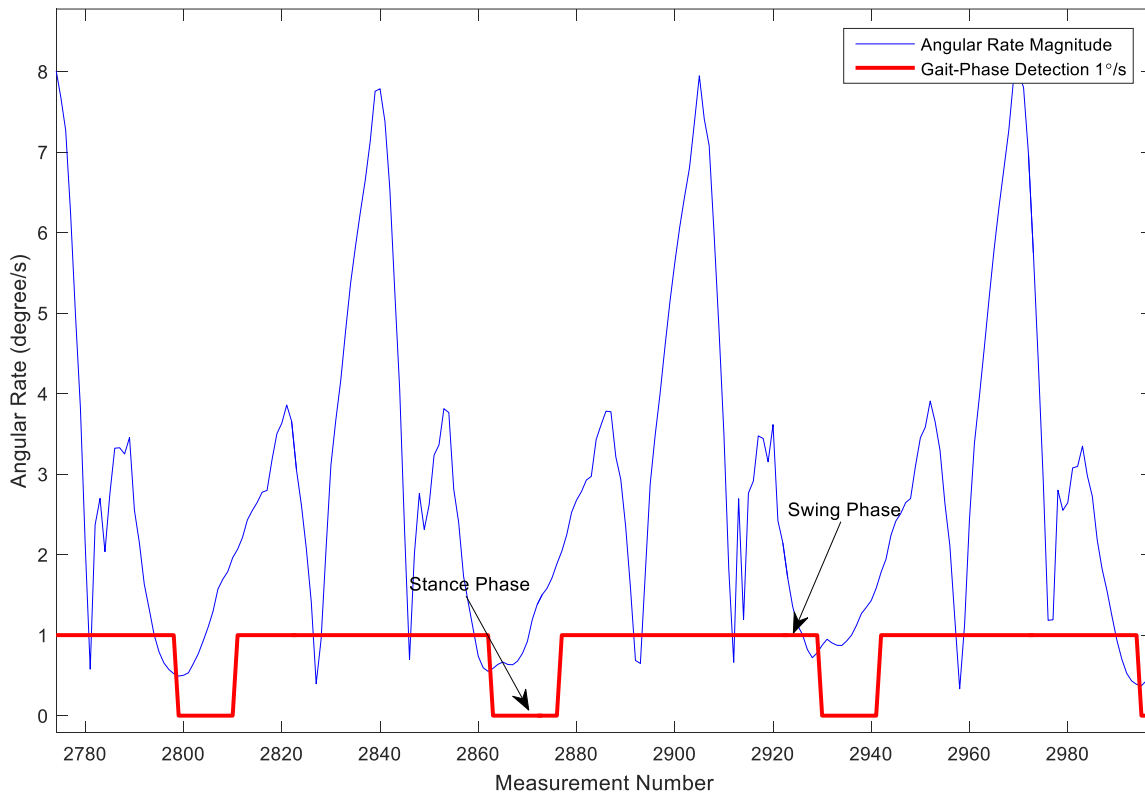


Figure 34. Combined Angular-Rate Length for Left Shin-Mounted IMU, Lap 10

## 2. The Lower Back

The lower back was assumed the best place to mount a sensor if it is true that dynamics degrade performance. The lower back is the inflection point about which most

movements occur; football coaches, for example, teach defensive players to focus on the ball carrier's hips (i.e., lower back) to estimate true path of travel. The biomechanics expert previously cited, suggested the T11 or T12 vertebral bodies specifically [66].

To characterize the performance of an IMU mounted on the lower back, the 2×1 bracket was used, secured by an elastic band around the user's waist. Sensors 1 and 2 were carried on the lower back for Laps 7 and 9. At the same time, Sensors 3 and 4 were mounted in a 2×1 bracket on the right foot. After processing Lap 9's lower-back data from through the PNS, a step-plot was created, as shown in Figure 35.

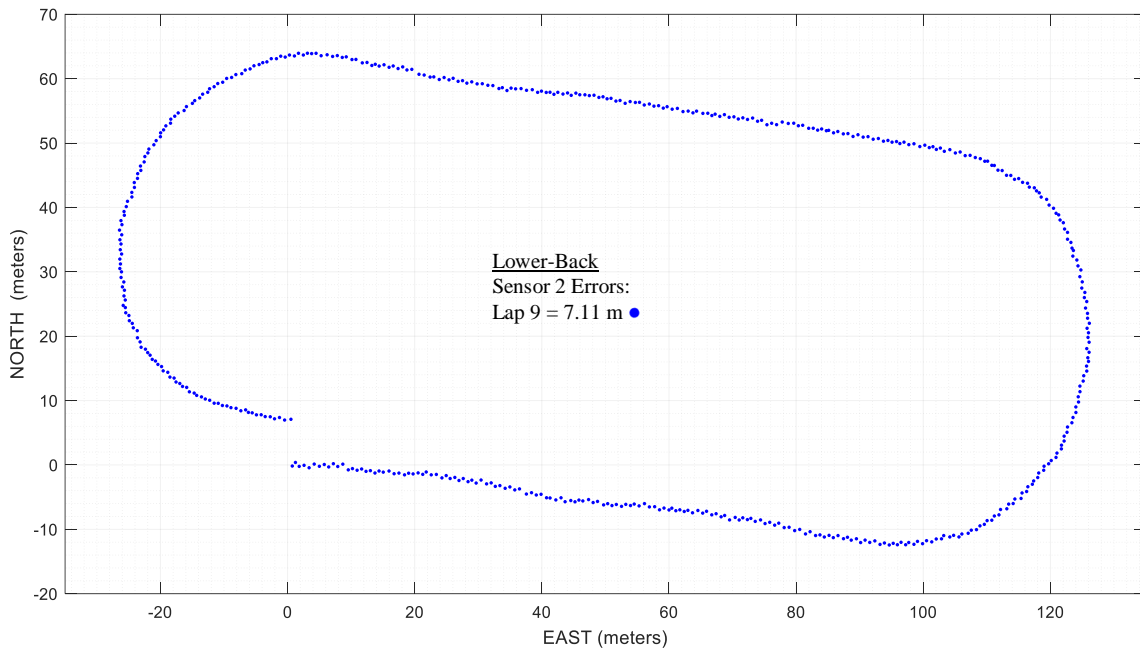


Figure 35. Lower-Back Mounted Step-Plot for Sensor 2, Lap 9; Scale Factor 12.137; Each Footstep (Left and Right) was Detected

Compared with the foot and shin locations, the first thing apparent with the lower-back data is the detection of twice as many footsteps. Since each blue dot represents a stance-phase detection (i.e., footstep), the data in Figure 35 clearly displays each step. The reason for both left and right footstep detection is the IMU placement. Foot swing and impulse forces are conducted up the leg, to the IMU strongly enough to identify gait-phases as well as measure forces. The complementary filter was not tuned to

work with data from both feet, and it took considerable effort to find settings that would produce an acceptable output (see Table 6).

Table 6. Optimal Settings for Lower Back Mounted Sensor 2, Lap 9

<b>Mounting Location: Lower Back</b>	<b>Value</b>
<b>Gait-Phase Detection Angular-Rate Threshold (deg/s)</b>	0.5
<b>Samples Above/Below Threshold Required to Change Gait Phase</b>	5
<b>Samples to Save</b>	20
<b>Complementary Filter Angular-Rate Threshold (deg/s)</b>	0.5
$k_s$	1.0
$k_d$	0.8
<b>Multiplied by Scale Factor</b>	12.137

Since the lower back endures nowhere near as much of an angular rate as a foot or shin, it was much more difficult to coax the gait-phase detection algorithm into effectively working. Fortunately, there were measureable differences between the swing and stance phases. The adaptive filter was adjusted so that the FQA remained a dominant source of data for both phases, as proven by  $k_s = 1.0$  and  $k_d = 0.8$  in Table 6. Like the step-plots from the shin-mounted sensor, Figure 35's step-plot for the lower back was scaled down. Unlike the shin's step-plot, however, the scaling reduction from lower-back measurements is dramatic, resulting in a total distance traveled of only about 10% of the track's true circumference. This scaling issue is rectified by incorporating a multiplication scale factor of 12.137 to the position data.

Another problem encountered with the lower-back mounted IMU is the small signal magnitude measured during the swing phases, which, when coupled with high noise in the stance phase results in a lower SNR as shown by the gyroscopes angular rate length in Figure 36. In this case, “noise” is the actual movement picked up by the IMU during the stance phase. Since the PNS assumes velocity during the stance phase to be zero, any non-zero measurement makes it more difficult for gait-phase detection logic to work correctly.

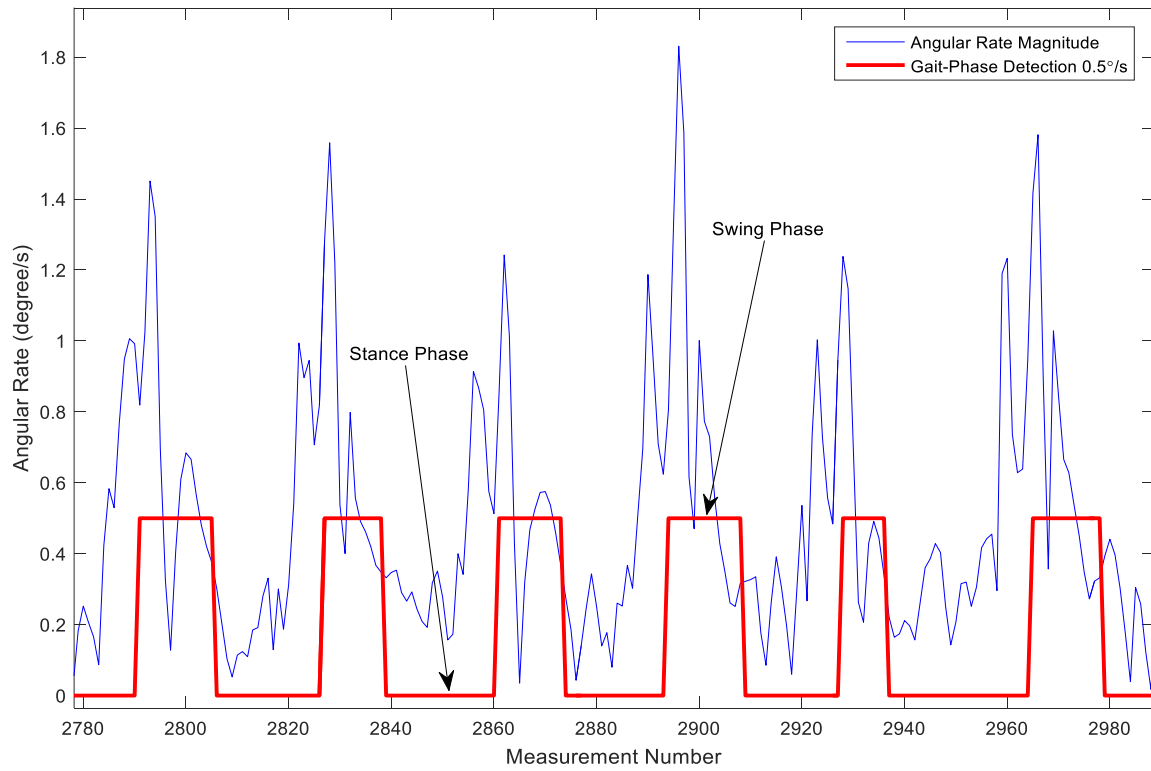


Figure 36. Combined Angular Rate Length for Lower-Back Mounted Sensor 2, Lap 9

As with the shins, Equation (3.13), which finds the norm of the  $X$ - and  $Y$ -axis was employed because Equation (3.14) worked poorly with lower-back data. Lower-back measurements, whose stance-phase measurements were typically near or above a magnitude of one resulted in an amplification of stance-phase noise instead of attenuation, as with the shins.

The gait-phase detection and measured gyroscope magnitude in Figure 36 demonstrates a low SNR of around 6 dB; therefore, mounting a gyroscope on the lower back for gait-phase detection is not optimal but can be done if desired. Despite the low SNR, the step-plot yields a reasonably shaped athletic track and, when scaled properly, has a moderate error of 7.11 m between starting and stopping points.

### **3. The Chest**

Moving away from the feet, the last location selected to characterize IMU performance was the chest. Much like the lower back, the chest detected each foot strike during walking, and also has the disadvantage of being more dynamic due to the natural rotation and bending of the torso during walking. Despite the difficulty of gait-phase detection and very active movement of the chest, the PNS functioned, albeit poorly, and might only be considered in applications requiring high accessibility to the IMU.

Of the locations tested, the chest gave worst performance. Requiring the old angular-rate formula (3.14), the chest produced a gait-phase SNR around 4 dB. This was less than the lower back's, and the chest's twisting and bending degraded it even more. To improve gait-phase detection, the weighting of the FQA was kept at a high level near 0.8 during the swing phase. This suggests the gyroscopes offered little enhancement to the PNS in swing phase as compared to the performance of the foot or shin. The step-plot of chest data shown in Figure 37 illustrates the reduced performance of the PNS at this location; note the noisy and jagged estimated path. A large-scale factor multiplier of 18.824 was required to achieve an accurate path length. After eliminating path compression, the error was lower than anticipated, 2.77 m, but without a nicely shaped path, this value means little and moreover may fluctuate significantly among test runs.

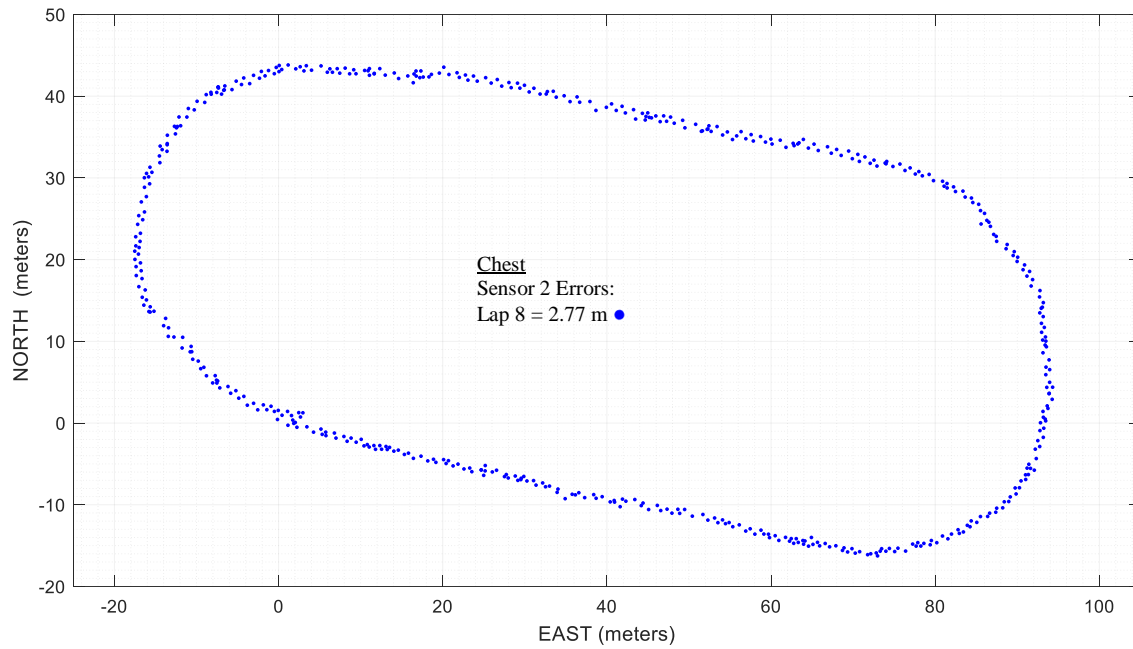


Figure 37. Chest Mounted Step-Plot for Sensor 2, Lap 8; Scale Factor 18.824;  
Each Footstep (Left and Right) was Detected

To achieve an adequately track shaped output, almost every setting was adjusted as shown in Table 7.

Table 7. Optimal Settings for Chest Mounted Sensor 2, Lap 8

Mounting Location: Chest	Value
Gait-Phase Detection Angular-Rate Threshold (deg/s)	0.253
Samples Above/Below Threshold Required to Change Gait Phase	3
Samples to Save	20
Complementary Filter Angular-Rate Threshold (deg/s)	0.253
$k_s$	1.0
$k_d$	0.8
Multiplied by Scale Factor	18.824

The gait-phase detection algorithm was barely capable of distinguishing between the stance and swing phases. In tuning the complementary filter for the chest, it was difficult to find a satisfactory angular-rate threshold value because changing this parameter by a thousandth of a degree-per-second resulted in missed steps or false detections. A gait-phase detection plot using data from the same test time as previous gait-phase detection plots is presented in Figure 38. The measured angular-rate data is low in magnitude compared with similar plots from the lower back, shins, and feet.

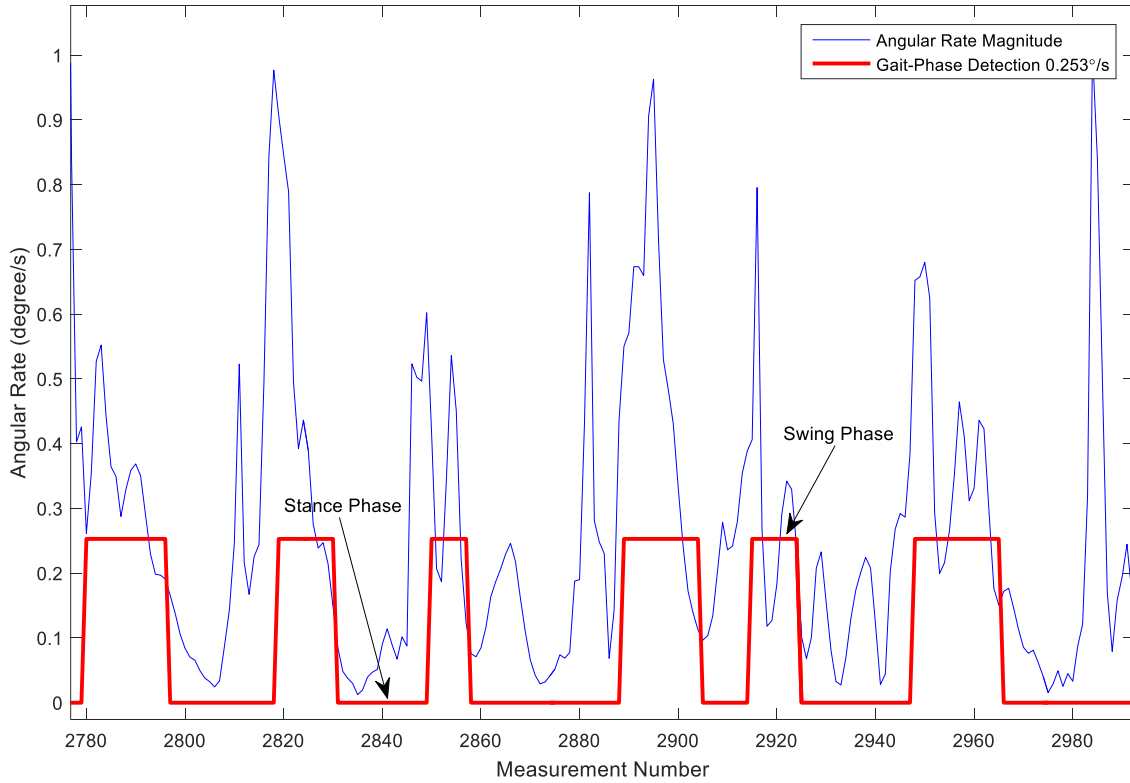


Figure 38. Combined Angular-Rate Length for Chest-Mounted Sensor 3, Lap 8

#### 4. Conclusion

The best place to mount a single IMU is the foot, owing to the high SNR of gyroscope measurements, which allows optimal gait-phase detection performance. The shin is second best but suffers from substantial movement during the stance phase owing to rotation about the ankle, making it more difficult to accurately detect gait-phase

changes. The lower back is third best, because it offers a stable platform but yields a lower gait-phase SNR as compared to the shin and foot. The worst performer is the chest. Less stable than the lower back, the chest has the lowest SNR of all tested locations, making gait-phase detection a challenge.

A clear trend emerges: as the IMU is mounted further away from the foot, PNS performance declines. The measured angular velocity during the swing phase is subsequently reduced, and the undesirable non-zero measurements during the stance phase are increased, resulting in scaling problems, gait-phase detection failures, and false alarms. By extrapolation, the ultimate mounting site for an IMU would be beneath the shoe. Since the sole is where the foot meets the ground, it experiences the greatest possible forces needed to calculate distance traveled. Mounting a sensor module on the instep resulted in consistent measurements of around 396–400 m. If the IMU were mounted to underside of a shoe (e.g., heel of a boot), after several runs the average path length might be very near the true track circumference of 404.6 m. If this location is deemed undesirable and the top of the foot is preferred, a small scaling factor of 1.02 may be used to multiply position output to account for the 10.0 cm or so distance between the top and bottom of the foot.

With respect to forces experience during human movement, the previously held assumption that higher accelerations and jerk forces degrade IMU performance was found to be false. Sensors within an IMU may slightly degrade due to the more intense dynamics of the foot, but compared to the sources of error seen in alternative mounting locations, these forces are insignificant. It is possible that these impulse forces may reduce the lifespan of an IMU, but testing to determine that potential reduction is outside the scope of this thesis.

## **E. DISTRIBUTED SENSORS**

The advantages and disadvantages of distributing unlike sensor types over a person's body, then fusing their measurements together into one virtual IMU before running data through the PNS algorithms are investigated in this section. The sensor types investigated in this research—gyroscopes, accelerometers, and magnetometers—

are conveniently included in the 3-Space module, and the knowledge gained from the characterization of IMU performance in this research can aid in the creation of a distributed system of sensors. “Distributed” is used to describe the spreading of different types of sensors over the body with the intention of improving PNS performance. For example, gyroscope data may come from a sensor module on the foot while accelerometer and magnetometer measurements derive from the lower back. The notion behind this investigation was that different types of sensors might perform better in different operating environments—for example, if the gyroscope works best on the foot and accelerometers and magnetometers work better on the lower back, separating these sensors will yield an overall improvement in PNS performance. The actual sensors within a 3-Space module were not separated for this experiment—they remained together. To achieve distribution, multiple 3-Space modules were used, but only the desired sensor data were read (e.g., the gyroscope at the foot and magnetometer at the lower back). In this way, it was possible to fuse the distributed sensor measurements taken from multiple IMUs into a single virtual IMU.

The results of sensor-performance characterization in this research suggested two ideal mounting locations for a distributed system. The first was the foot, which provides the best gyroscope performance by enabling superior gait-phase detection with high SNR and few scaling problems. Right-foot mounting was selected as a standard for convenience (see Figure 21) in view of the right-dominance of most users including the tester.

It was hypothesized that the optimal location to mount an accelerometer or magnetometer would be the lower back because of its stability and low-dynamics. The IMU was secured onto the lower-back using the 2×1 mount and elastic bands as exhibited in Figure 39. Though research showed that the IMU performs best when mounted on the foot, it was conjectured that most of these benefits were due to the improved gait-phase detection and dynamic measurements, both made by the gyroscope, and that perhaps the jerk forces experienced by the accelerometer and magnetometer during a heel striking the ground degraded their performance. Improving the pitch and roll estimates made by the

accelerometer, as well as the heading estimates from the magnetometer, might boost FQA, and overall PNS performance, resulting in higher position accuracy.

Chest mounting was ruled out for its instability and poor gait-phase detection, and the feet were preferred over the shins because of their superior gait-phase detection and low dynamics in the stance phase, which are needed to provide the best ZUPTs. For these reasons, distributed testing in this research focuses solely on sensors mounted on the right foot or lower back.



Figure 39. 2×1 Bracket Holding Sensors 1 and 2 on Lower Back

### **1. Selection of Reference Frame: Body or NED**

It was assumed that for a distributed system to work, all sensors require their data to be transformed into an NED reference frame, also called a navigation frame. This is because sensors distributed over the body measure forces with respect to themselves; they do not know the orientation of other sensors and therefore cannot translate forces into a coherent picture. If a force is registered entirely in the  $X$ -axis on a lower-back mounted sensor, then this force may align with a foot-mounted sensor's  $Y$ -axis direction one moment and with the  $Z$ -axis another, causing a disparity in measurements. It was assumed that when this mismatched data was combined, the PNS would output gibberish instead of, for example, providing a step-plot in the shape of a track. Tests proved this

assumption somewhat inaccurate. Body-frame gyroscope data from one foot-mounted sensor can be appended to NED accelerometer and magnetometer data from a sensor on the lower back and have exactly the same performance as if the gyroscope had been transformed to NED. The reason may be due to unlike sensor types measuring different types of forces (e.g., angular velocity versus linear acceleration) or the way the complementary filter processes these forces. The dynamic branch uses the angular rate to determine distance traveled, while the static branch uses linear acceleration primarily to determine pitch and roll. The main source of this acceleration is gravity. To detect position, the dynamic branch relies not so much on orientation as it does angular-rate forces. Further investigation demonstrated a need for accelerometer data to be converted to a local NED coordinate system before combining it with other sensor types (i.e., gyroscope or magnetometer).

A MATLAB GUI was created to simplify repeated transformations to NED, as seen in Figure 40.

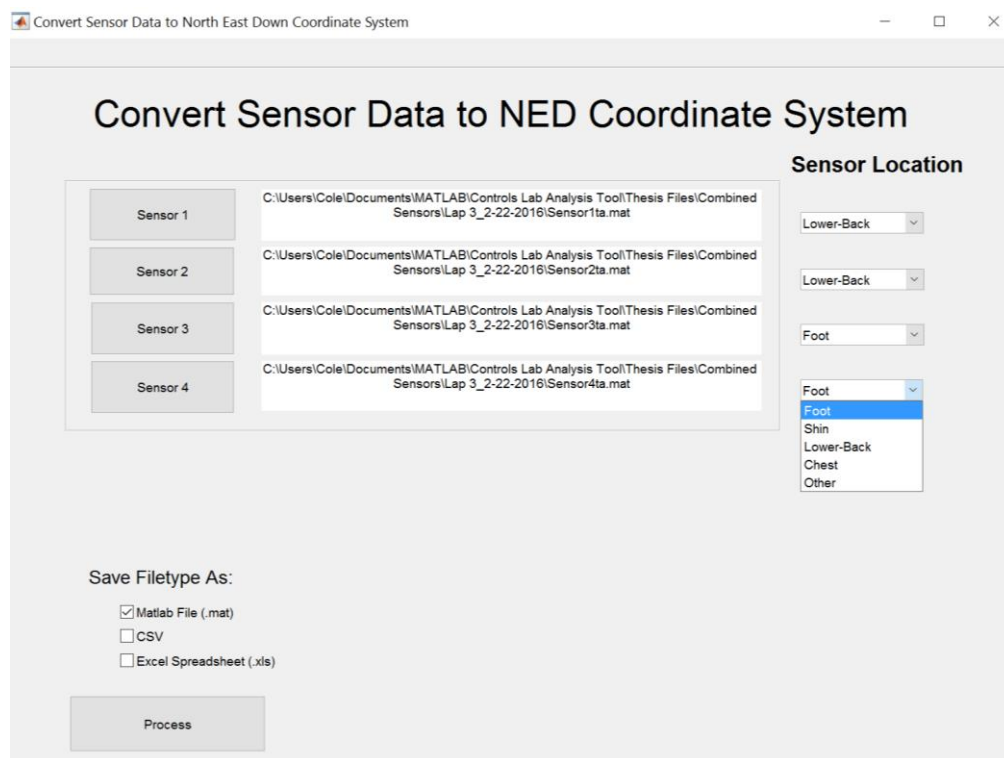


Figure 40. GUI to Convert Body Frame Data to NED Coordinate System

Surprisingly, converting the gyroscope and/or magnetometer to a NED coordinate system or leaving them in a body-referenced frame returned similar PNS performance. Overall, the best PNS performance was achieved when all sensor types were converted to a NED coordinate system before combining measurements into a single data file; therefore, further distributed test measurements were converted to a NED coordinate system before combining data from multiple sensors.

## 2. Optimal Mounting Locations for Gyroscope, Accelerometer, and Magnetometer

The next step was to determine the optimal mounting locations for the gyroscope, accelerometer, and magnetometer. The initial assumptions that predicted the foot as optimal for the gyroscope were borne out in testing. The accelerometer and magnetometer were assumed to achieve their best performance when mounted on the lower back since their job is to determine roll, pitch, and heading with the FQA.

Before plotting distributed-sensor data, a baseline plot was created from the foot-mounted Sensor 3. These data are displayed in Figure 41.

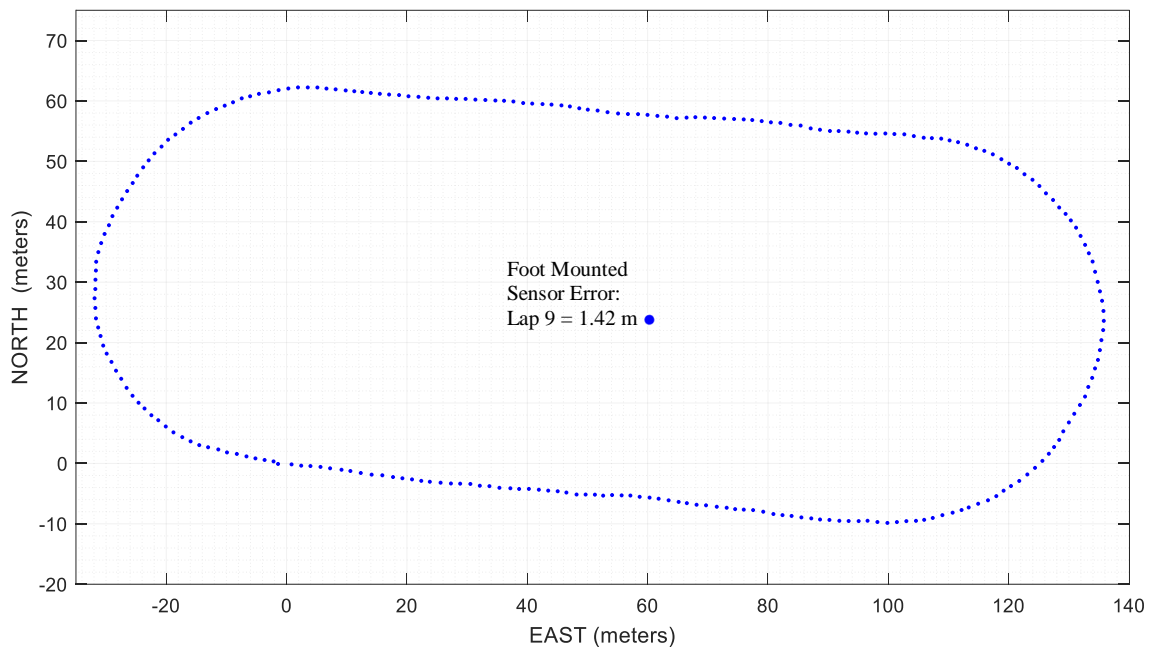


Figure 41. Step-Plot Displaying Baseline Reference using Sensor 3's Foot-Mounted Measurements from Lap 9; Scale Factor 1.01

The position output was multiplied by a value of 1.01 to increase the total distance traveled from 401.6 to 404.6 m, the circumference of the track. Subsequent step-plots were also scaled to produce a total traveled distance within 0.5 m of the circumference. Normalization of step-plots is performed in this section to allow for comparison regardless of sensor mounting location.

***a. Gyroscope Mounted on Foot, Accelerometer/Magnetometer on Lower Back***

The first test analyzed sensor data using the assumed optimal configuration, where gyroscope measurements were derived from a foot-mounted sensor, while accelerometer and magnetometer measurements came from the lower back. The resultant plot, as presented in Figure 42, shows significant scaling errors.

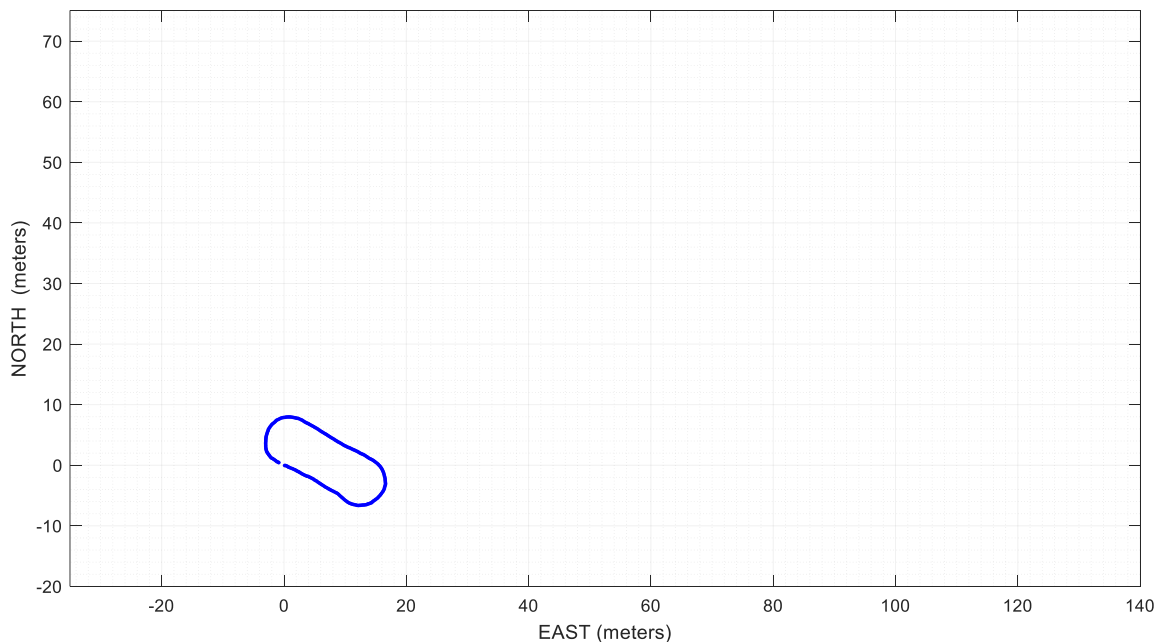


Figure 42. Step-Plot with Unadjusted Scale and Direction, Lap 9; Foot-Mounted Gyroscope, Accelerometer/Magnetometer Back; Scale Factor 0.0

The scaling errors were very similar to that found and discussed previously. In addition, for some unknown reason, the direction of the plot is rotated clockwise by several degrees. It is unknown why the rotation occurred, but later plots show the

placement of the magnetometer does not influence performance. It is not thought that an orientation change was the cause either. This is because before combining measurements from multiple IMUs mounted on different body parts in a single virtual IMU, each sensor has its measurements converted to the navigation frame. Despite these issues, the general shape of the plot is recognizable as the track where the test took place. The gyroscope data in Figure 42 is derived from a foot-mounted Sensor 3, while the accelerometer and magnetometer data is from a lower-back-mounted Sensor 2.

To adjust the scale of the step-plot in Figure 42, a multiplication factor was empirically found.

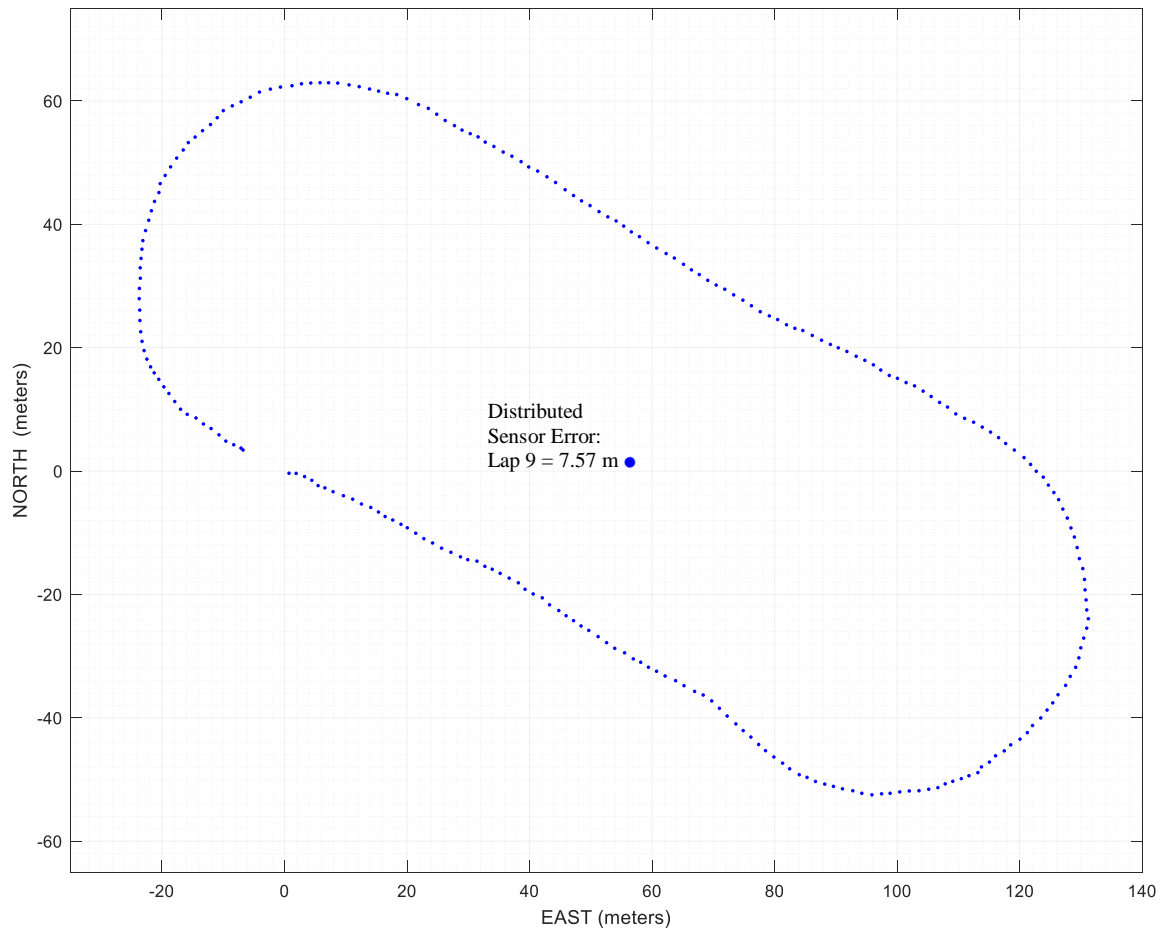


Figure 43. Step-Plot with Scale Adjusted, Lap 9; Foot-Mounted Gyroscope, Accelerometer/Magnetometer Lower Back; Scale Factor 7.9

For this particular test, the output position is multiplied by 7.9, and the resulting total distance traveled was then calculated to be 404.31 m, which is very close to the true one-lap circumference of 404.6 m. The resultant error indicates the difference between the starting and ending location is 7.57 m. The properly scaled step-plot for this test run is presented in Figure 43. The overall size and shape match the test track. Like the shrunk step-plot in Figure 42, there is a significant heading error in the fully scaled Figure 43.

For visual comparison, the high-performing baseline step-plot, which contains only foot data from Sensor 3 on the same lap (i.e., Lap 9), was overlaid with the current distributed sensor step-plot from Figure 43; the result is shown in Figure 44.

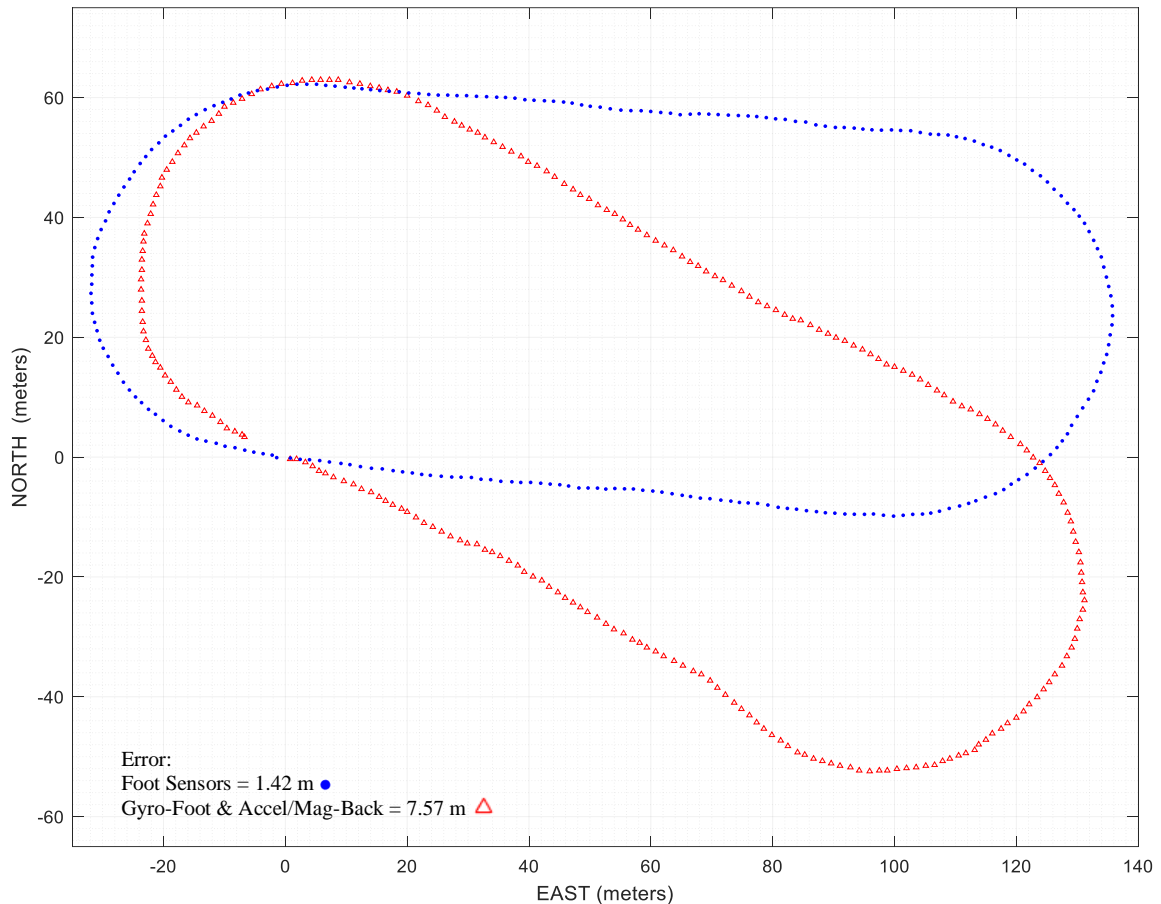


Figure 44. Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Gyroscope on Foot and Accelerometer/Magnetometer on Lower Back

Now the differences become clear, indicating that foot-mounting the gyroscope and back-mounting the accelerometer and magnetometer sensor do not improve PNS performance. Rather, this distribution slightly degrades performance, even after implementing a scaling factor and tuning gains optimized for each body location. Small performance differences between sensors do not account for the degradation. Moreover, quite a bit more work is required to synchronize the sensors, which further discredits this arrangement. Time-synchronization techniques used to align data are discussed later.

***b. Accelerometer on Foot, Gyroscope/Magnetometer on Lower Back***

The plot data presented in Figure 45 sources accelerometer data from the foot, while gyroscope and magnetometer data come from the lower back.

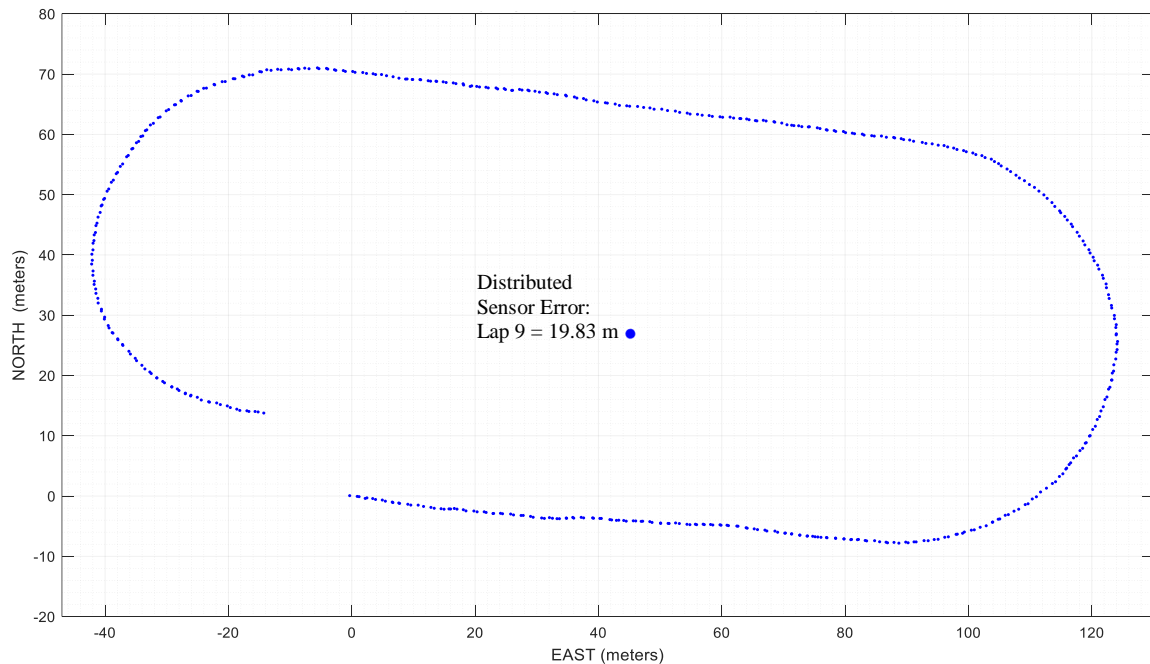


Figure 45. Step-Plot with Foot-Mounted Accelerometer, Gyroscope and Magnetometer from Lower Back; Lap 9, Scale Factor  $-2.54$

A scaling factor of  $-2.54$  is required to correct the scaling and orientation of the plot. For reasons unknown, if the scaling multiplier is not negative, the resulting plot is flipped and reversed. Because the gyroscope was mounted on the lower back, the

gait-phase detection algorithm uses a lower SNR to accurately determine stance- versus swing-phase, and all footsteps (both left and right) are detected. The accelerometer data in Figure 45 is derived from a foot-mounted Sensor 3, while the gyroscope and magnetometer data is from a lower-back-mounted Sensor 2.

The performance of this particular sensor distribution is compared to foot-only data from the baseline step-plot and is shown in Figure 46. Lap 9's distributed arrangement performed worse than Sensors 3's foot-only measurements. Deriving all sensor data from the lower back, whose data is shown in Figure 35, we get better performance than the distributed setup. Lower-back-only data had a 7.11-m error, significantly more accurate than the 19.83-m error of this distributed sensor setup. Once again, the best performance was achieved by processing foot-only IMU data.

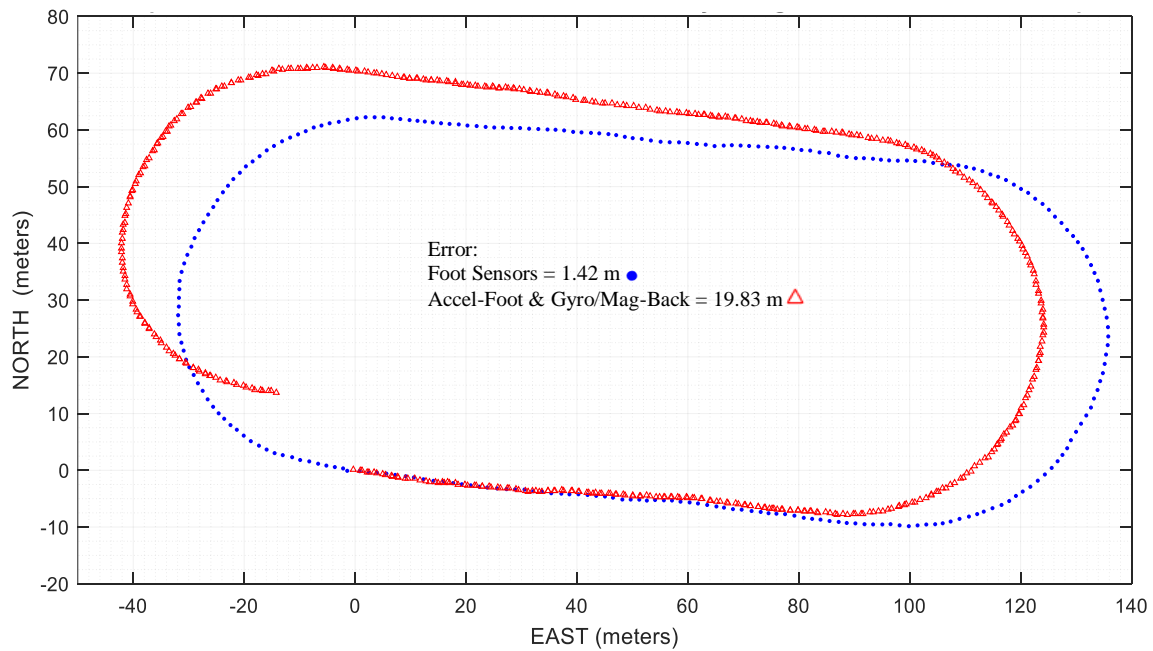


Figure 46. Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Accelerometer on Foot and Gyroscope/Magnetometer on Lower Back

*c. Magnetometer on Foot, Gyroscope/Accelerometer on Lower Back*

Magnetometer measurements recorded on a foot-mounted IMU are analyzed and the result with three arrangements of gyroscope and accelerometer mounting locations are paired in this subsection. This was done to consolidate the poorest-performing distributed arrangements into one section. All data was taken from Lap 9.

The first two distributed setups alternated the gyroscope and accelerometer locations. All three were bad performers. In the initial setup, gyroscope data came from the right foot and accelerometer data from the lower back. The second setup was opposite: gyroscope data from the lower back and accelerometer data from the right foot. The third setup provided the best results, with magnetometer data from the right foot and gyroscope and accelerometer data from the lower back. However, compared to previous tests the results were poor.

It quickly became apparent that separating gyroscope and accelerometer sensors results in extremely poor performance, as exemplified by their step-plots' not resembling an oval track. The data appeared to be randomly scattered throughout their own step-plots, which are excluded from this thesis.

The only combination in which the PNS was able to produce a step-plot that resembled an oval track, however crudely, was when the gyroscope and accelerometer were mounted to the lower back and the magnetometer was mounted on the foot. This data is represented by red triangles in Figure 47. Complementary filter settings were set to Table 6 values, which had previously been found to work well with the lower back. The scaling of the resulting step-plot was similar to lower-back gyroscope or accelerometer measurements. Despite the reduced performance, every step taken by either foot was properly detected.

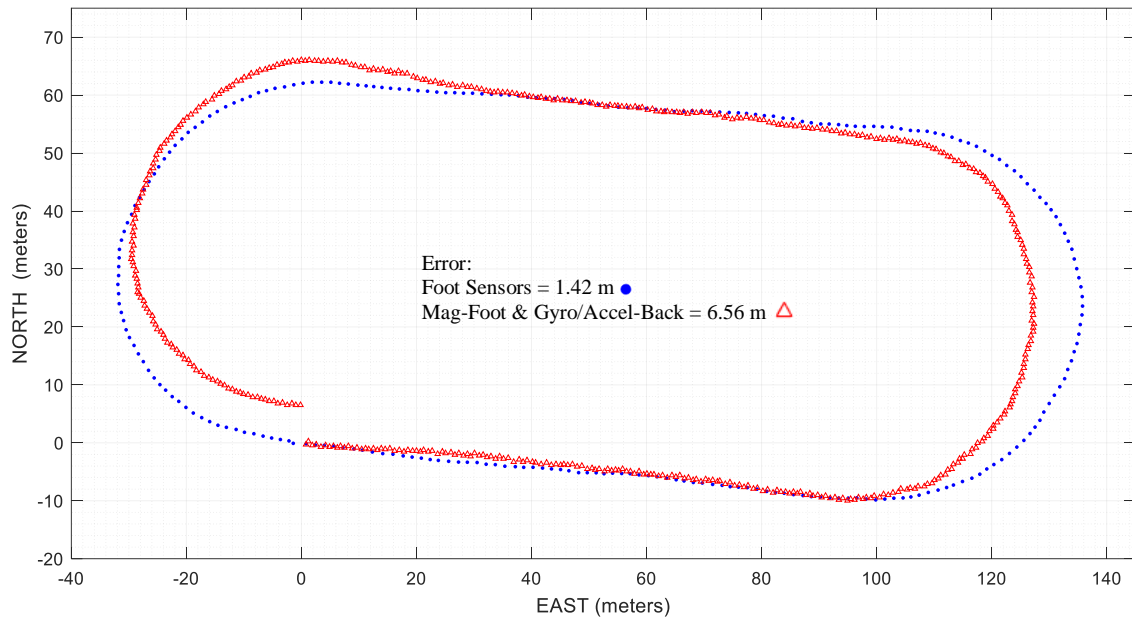


Figure 47. Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Magnetometer on Foot and Gyroscope/Accelerometer on Lower Back

Placing the gyroscope and accelerometer in the same lower-back location, with the magnetometer on the foot, we got better performance than when the gyroscope and accelerometer were split up. This setup shows exactly the same performance and step-plot output as when processing only Sensor 2's measurements taken from the lower back (see Figure 35).

Since the PNS performance was found to be exactly the same as with the lower back when mounting the magnetometer on the foot, changes in performance based on magnetometer location were negligible. Most operating environments would benefit from sourcing magnetometer-heading measurements from the same sensor module as the gyroscope and accelerometer. An exception may be when the gyroscope and accelerometer mounting locations bring the magnetometer physically closer to magnetic interference from ferrous materials or current flowing through wires. If separation of the magnetometer from the rest of the sensor is required, note that the magnetometer is less sensitive to time misalignments than the other two sensors; synchronizing time with this arrangement is not as important as it would be if the gyroscope and accelerometer measurements came from different IMUs.

***d. Gyroscope/Accelerometer on Foot, Magnetometer on Lower Back***

Two optimal distributed configurations of sensor types were found. Both occur when the gyroscope and accelerometer data are sourced from the foot. The origin of magnetometer data makes no difference in resulting accuracy, as illustrated by the data in Figure 48. This enables the user to use a preferred mounting location for the magnetometer without sacrificing performance. No difference was seen whether the complementary filter used the foot settings in Table 4 or lower-back settings in Table 6.

The foot clearly emerges as the optimal location to mount a sensor module. There was no added benefit to performance by mounting the magnetometer on the lower back while the gyroscope and accelerometer measurements were on the foot. The best possible scenario is to measure gyroscope, accelerometer, and magnetometer data from the same foot mounted IMU. This reduces complexity by allowing the user to omit manually synchronizing the magnetometer measurements with the gyroscope and accelerometer measurements, because all data will have already been synchronized within the IMU.

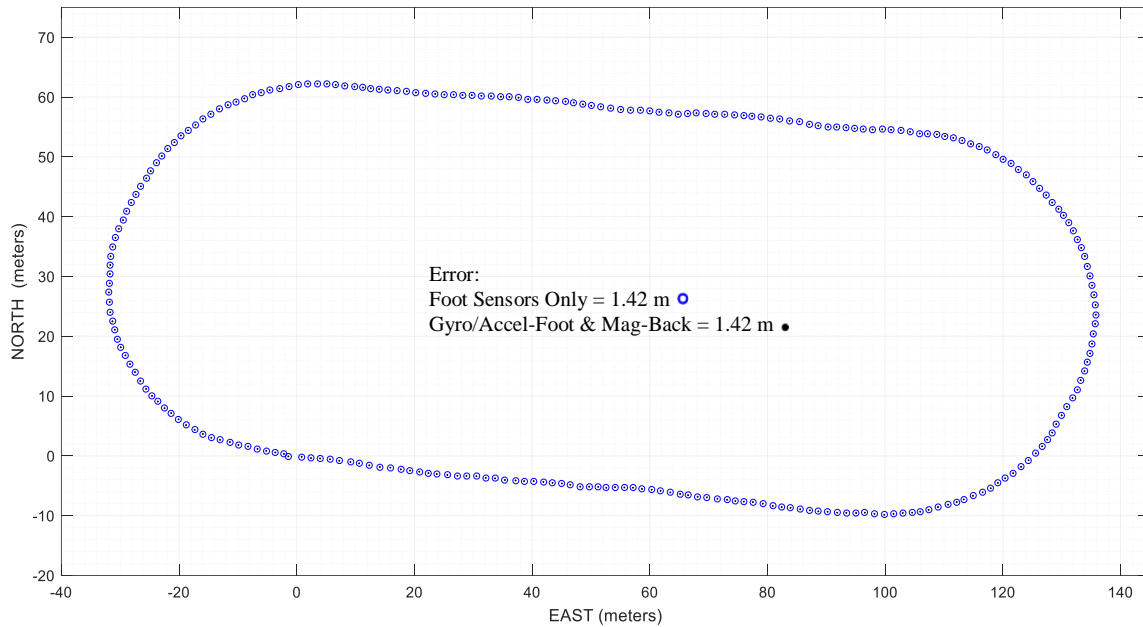


Figure 48. Two Step-Plots: Foot-Mounted IMU versus Distributed IMU with Gyroscope/Accelerometer on Foot and Magnetometer on Lower Back

*e. Average Multiple Sensors Together Before Combining*

Since there was a large performance difference between distributing sensors over the body versus using a single foot mounted IMU, it was assumed that there would be little benefit in doubling up the number of distributed sensors—in other words, the performance degradation caused by a poor mounting location would not be mitigated by adding additional IMUs to the same location. To test this assumption, two sensors from each location (the foot and lower back) had their measurements averaged together. Averaging was possible because the sensors could be mounted together in a 2×1 bracket. The arrangement for the lower-back-mounted Sensors 1 and 2, is shown in Figure 49, and the foot-mounted Sensors 3 and 4 in Figure 50.



Figure 49. 2×1 Module Bracket with Sensors 1 and 2, Mounted on Lower Back



Figure 50. 2×1 Bracket with Sensors 3 and 4, Mounted on Right Foot

Each set of two co-located sensors were averaged together using techniques discussed later. The first attempt at averaging two sensors resulted in performance similar to that of a single sensor, as revealed in Figure 51; thus, there is no benefit to averaging two sensors if the mounting location is poor.

This particular setup sourced Lap 9's gyroscope data from the foot, while accelerometer and magnetometer data was taken from the lower back. A scale factor of 7.9 was required to normalize the size of the plot.

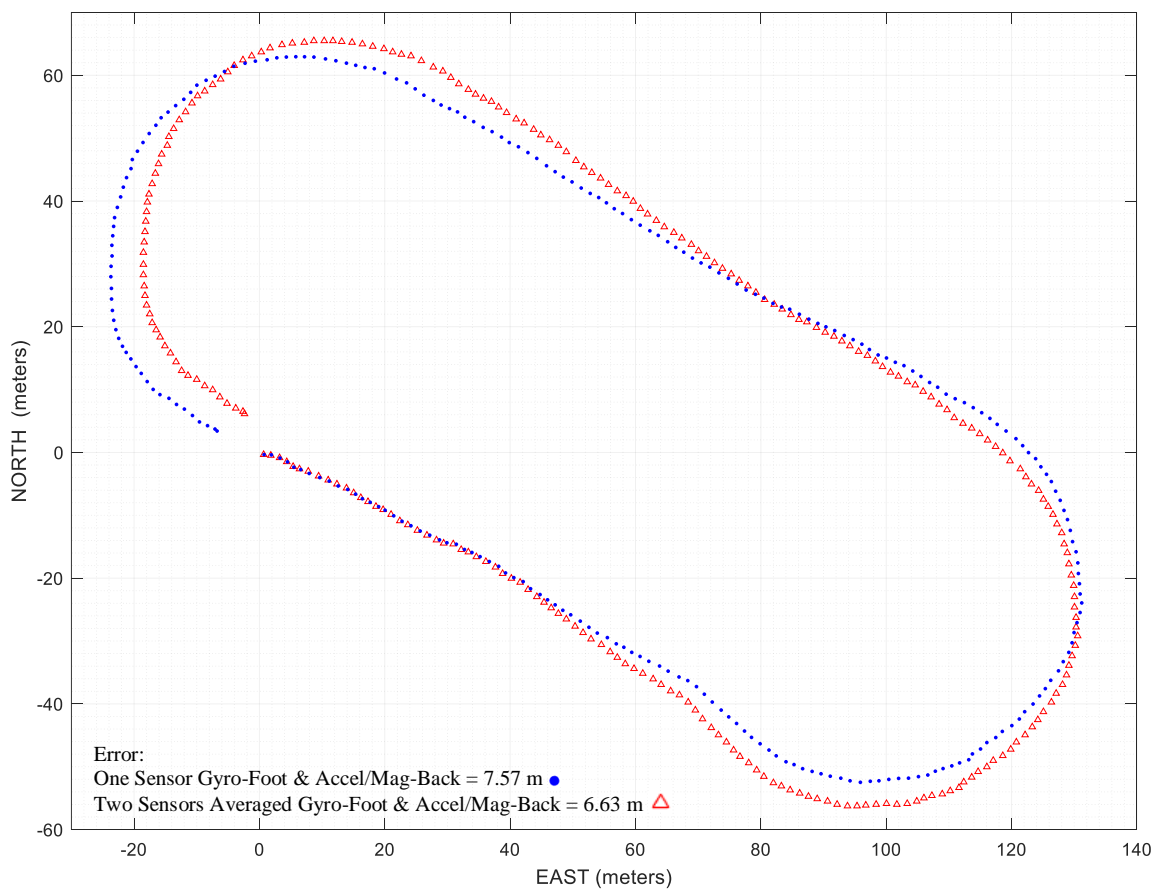


Figure 51. Averaging Two Distributed Sets of IMUs in Poor Mounting Locations versus One IMU in Same Locations

The same comparison was made with a slight configuration change: two sets of accelerometers were mounted to the right foot and two sets containing both gyroscopes and magnetometers were attached to the lower back. The data were then averaged as

before and is provided in Figure 52. Like Figure 51, there was no significant performance improvement obtained by averaging two distributed sets of two, co-located sensors in a poor location when compared to a single set.

The data was again taken from Lap 9, but two slightly different scale factors were required to normalize the plot sizes. The averaged two distributed sensors required a scale factor of  $-2.54$ , while the single distributed sensor needed  $-2.32$ .

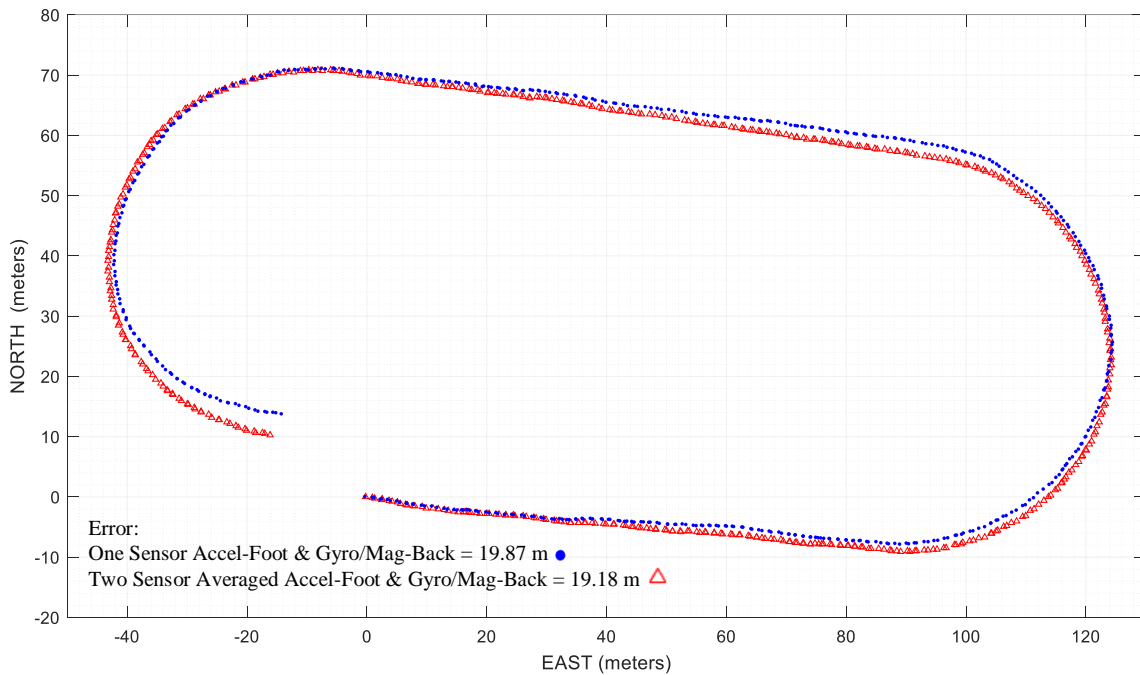


Figure 52. Averaging Two Distributed Sets of IMUs in Poor Mounting Locations versus One IMU in Same Locations

Since there was no performance difference between magnetometer measurements from the foot or lower back, no need was seen to average the two sets of magnetometer measurements together. Averaging the measurements from sets of two, three, and four co-located IMUs is addressed later.

*f. Conclusion: The Foot is the Best Location for All Sensor Types*

Mounting both the gyroscope and accelerometer on the foot offers significantly better performance than mounting them on the lower back. For the magnetometer,

mounting on the foot or lower back yields identical performances. When low-cost, compact IMUs improve their raw specifications by at least a magnitude, the necessity of using a magnetometer for heading measurements will be eliminated, but further complexities will likely arise, such as the need to perform an initial gyroscope alignment after applying power. This alignment will take a few minutes to complete, because the gyroscope is refining its alignment by detecting the spin axis of the earth and subsequently, true north. Over time, the true heading error would steadily increase based on the specifications of the unit.

Averaging two co-located sensors of the same type does not improve performance enough to compensate for a suboptimal mounting location. In other words, if there is a requirement to mount an accelerometer or gyroscope to the lower back or chest, placing additional sensors at the same location does not overcome the performance degradation expected.

#### **F. AVERAGING OF MULTIPLE FOOT-MOUNTED SENSORS**

The benefits of mounting multiple sensors on the same foot are investigated. The 2×2 sensor bracket, which holds four sensor modules in a constant mutual orientation (or orthogonal redundancy), was configured on the foot as demonstrated in Figure 23. This redundant configuration is an attempt to execute a concept originating with [56], which calls for manufacturers to create an array of four gyroscopes printed on the same MEMS board within millimeters of one another. Research in the topic of MEMS IMUs suggests that the low yield rates (~3%) of sensor manufacturers prevent them from manufacturing this four-in-one design, despite substantial performance benefits [67].

The outputs of the 3-Space modules were averaged together in groups of two, three, and four before processing the measurements through the PNS. Averaging multiple sensor modules into a virtual IMU (VIMU) was found to offer a modest performance increase of 37.7% [55]. This research also finds that using the least-squares method to combine multiple sensors offered only 2.4% better performance than averaging them together, and a more complicated adaptive Kalman filter offers only 4.4% better performance than averaging. It was decided that using the least-squares and Kalman-filter

methods of combining multiple sensor measurements into one output does not overcome the sharp increase in project complexity and computational demand required; the minor performance boosts provided are not worth the substantial effort required to employ them. As a result, multiple sensors were only averaged together.

The sensor measurements in this section are all derived from the same test lap (i.e., Lap 4) to eliminate extraneous variables. The sensors were arranged within the 2×2 bracket as exhibited in Figure 53 and mounted on the right foot as shown in Figure 23.

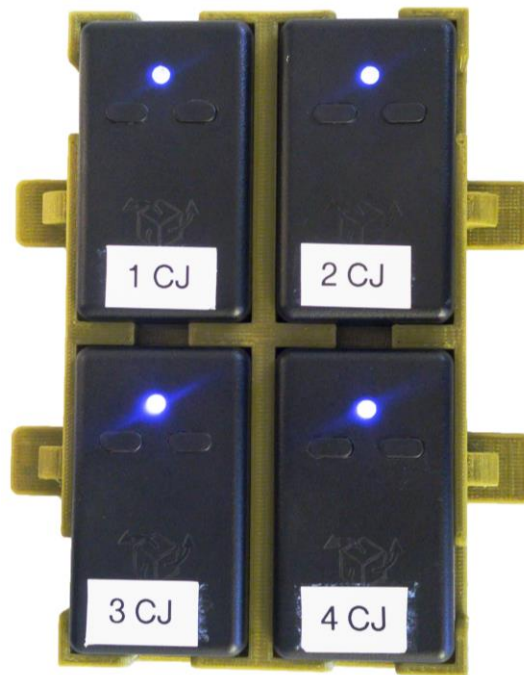


Figure 53. Orientation of Sensors 1–4

### 1. Sample-Rate Inconsistencies

Before the sensor measurements were combined, great care was taken to ensure the recorded outputs were truly aligned. Though each sensor's clock was synchronized to the same laptop computer and calibrated minutes before testing, their times were not well aligned. The plotted angular rates from all four sensors, which use the exponential formula (3.14), are shown in Figure 54. Note that the same physical force felt by two of the four sensors is separated by almost 40 samples despite time synchronization of each

through software. In reality, those forces occurred simultaneously. To complicate matters further, despite trimming each data file to align beginning-of-lap data, the sampling rates for each sensor were still unique enough to cause the sensors to drift out of synchronization.

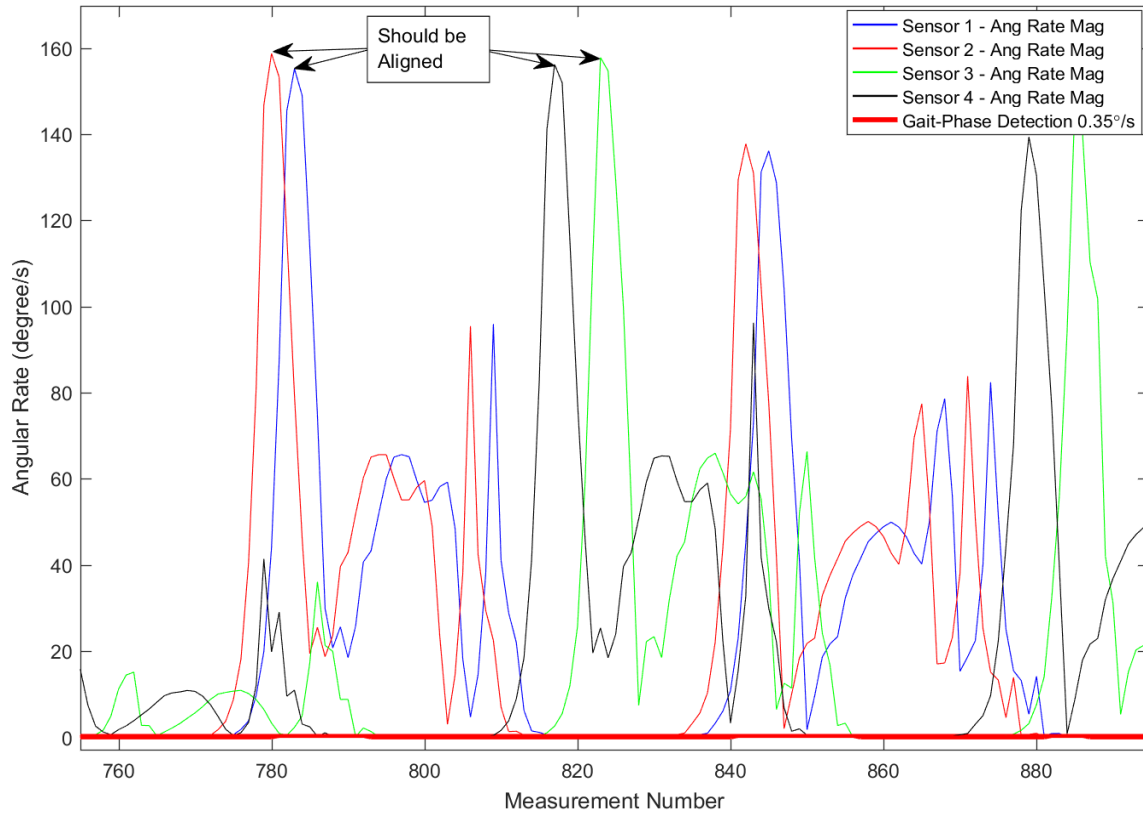


Figure 54. Misaligned Angular-Rate Length Data of Four Time-Aligned IMUs

To properly time-align the four data sets, 26 post-processing corrections were manually applied to correctly align the four sensors' measurements. This was a lengthy process, even though the data were from one test lap that only took five minutes to complete. Aligning all four of the sensors, as demonstrated by the data in Figure 55, was accomplished by laboriously monitoring the alignment of angular-rate length peaks. When it was noticed that a specific sensor's angular-rate length peak was out of synchronization, segments of data, usually one or two samples' worth, were removed from the offending sensor to maintain alignment. Since the data sets from each IMU

contained over 19,300 samples, the removal of 26 samples to achieve alignment was considered acceptable. Without doing so, characterizing the benefits of averaging multiple sensors together was impossible.

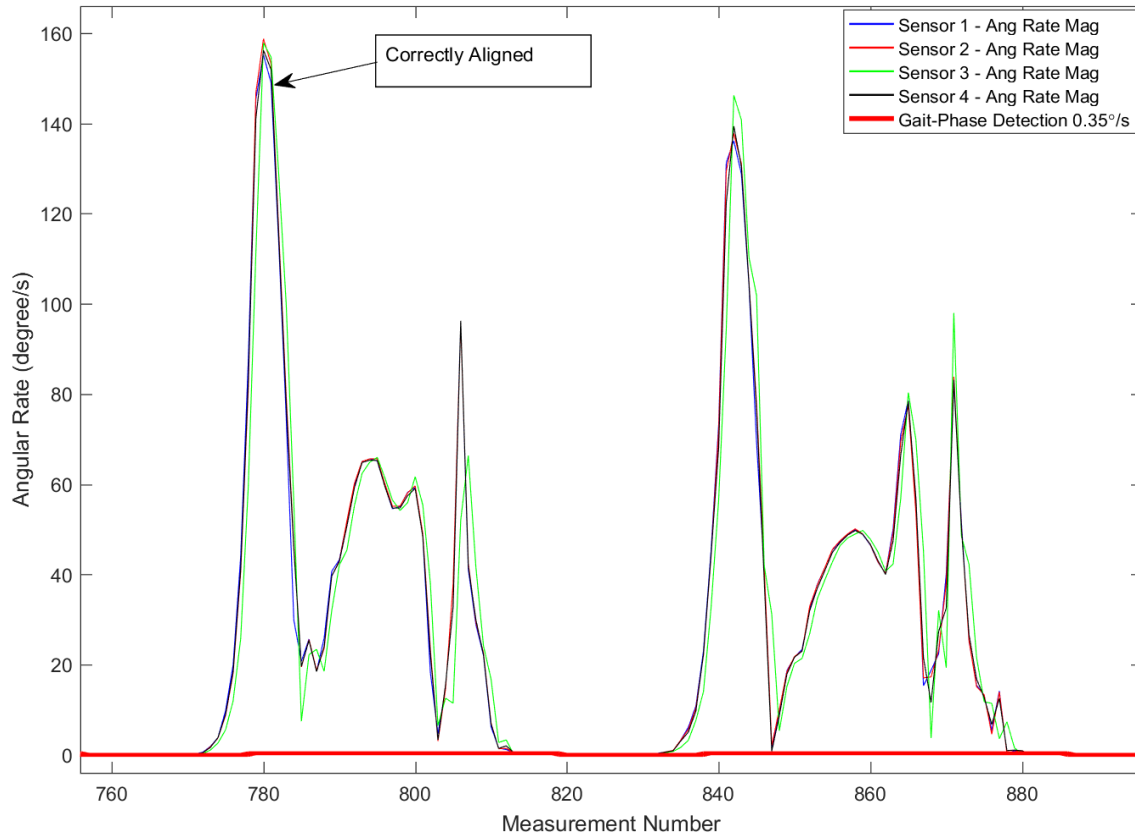


Figure 55. Same Angular Rate Length Data as in Figure 54, Except Measurements Manually Aligned

Further investigation of IMU alignment problems revealed that the units did not consistently sample at the expected rate. Based on system documentation, when the 3-Space sensor module's capture interval is set to automatic, it is expected that each sensor chooses the sampling frequency that offers the highest sample rate that can be consistently maintained [42]. To examine sample-rate inconsistencies for the four sensors, data from the one-hour, forty-minute PSD runs were used, as plotted in Figure 56. The two dark horizontal lines demonstrate that the vast majority of sample frequencies for the four IMUs were 61.5 Hz or 63.3 Hz. Sensors 2 and 4 dropped their

sample rates to 45 Hz and 25 Hz, respectively, on a consistent but unpredictable basis, whereas 1 and 3 did not. This sudden drop in sampling frequency persisted when settings were adjusted to 50 Hz as well as when set to the IMU's largest allowable sample-rate settings of 240 Hz. The sample rates switched between 215 Hz and 240 Hz about as often as with lower sample frequencies. Due to the inaccuracies of each IMU's onboard clock, it is impossible to utilize time as an effective method to eliminate data misalignment caused by sample rate changes between sensors.

These results indicate that to achieve accurate runs of the PNS using multiple IMUs in real time, a robust method to account for sampling rate variations and ensure data synchronization is needed. Without such an ability, the benefits of using multiple sensors are lost and performance degrades to a level worse than that of a single IMU.

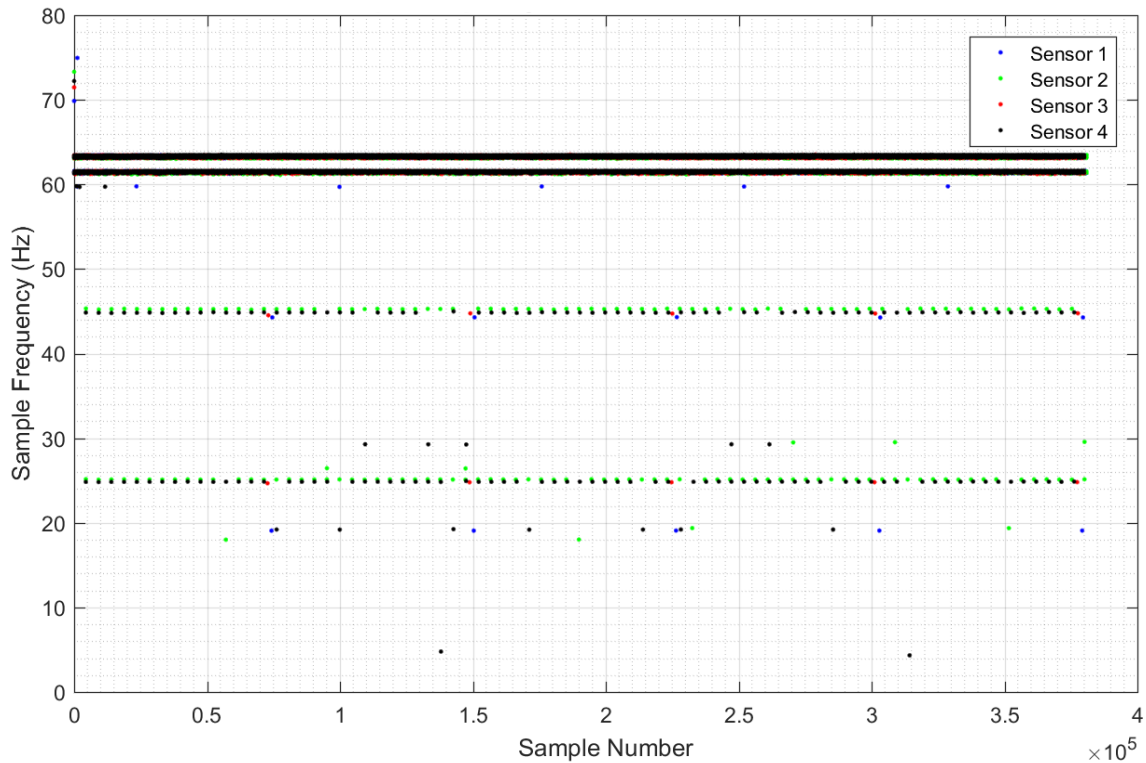


Figure 56. IMU Sample Frequencies Displaying Inconsistencies Despite Identical Settings

For the Reticle system and other applications, the synchronization of data through a wireless connection is considered unsuitable because each sensor must emit and receive radio frequency (RF) signals to process the data. One of the main advantages of IMUs in the field is that they are passive. By contrast, a device that emits RF energy in a war zone puts the operator at risk of detection. When multiple IMUs communicate through an RF data-link, their transmissions may be intercepted and warn the enemy of the user's encroachment. The user is also vulnerable to communication jamming through unintentional or intentional destructive interference from an RF noise source, rendering the PNS useless. (Note that some applications require an RF signal sent by other users in the form of a datalink. This is different from relying on an RF signal for proper PNS operation and is acceptable.)

Nevertheless, there are other ways to perform synchronization. One is to purchase hardware with a more accurate clock and stable sampling rates. Another is to install a wired synchronization system. Since averaging multiple IMUs at the same location requires close mutual proximity, wiring them together may be trivial.

## **2. Averaging Two Sensors**

After receiving proper alignment, two 3-Space sensor modules were averaged to determine if there were advantages over a single sensor module. The four IMUs were arranged in sets of two, and a few different pairings were tested to determine consistency of performance and the amount of complementary-filter tuning required. Each combination was averaged using a GUI built for the purpose, as depicted in Figure 57. The GUI executes an algorithm that averages the parameters from one 3-Space module with the corresponding columns and rows of another. For example, if Sensors 1 and 2 are loaded, the logic takes the data from [1, 1] of Sensor 1 and averages it together with the data from [1, 1] of Sensor 2. As noted previously, this algorithm is only effective when each IMU has its data sufficiently aligned with the other IMUs. Otherwise, performance is degraded, not improved.

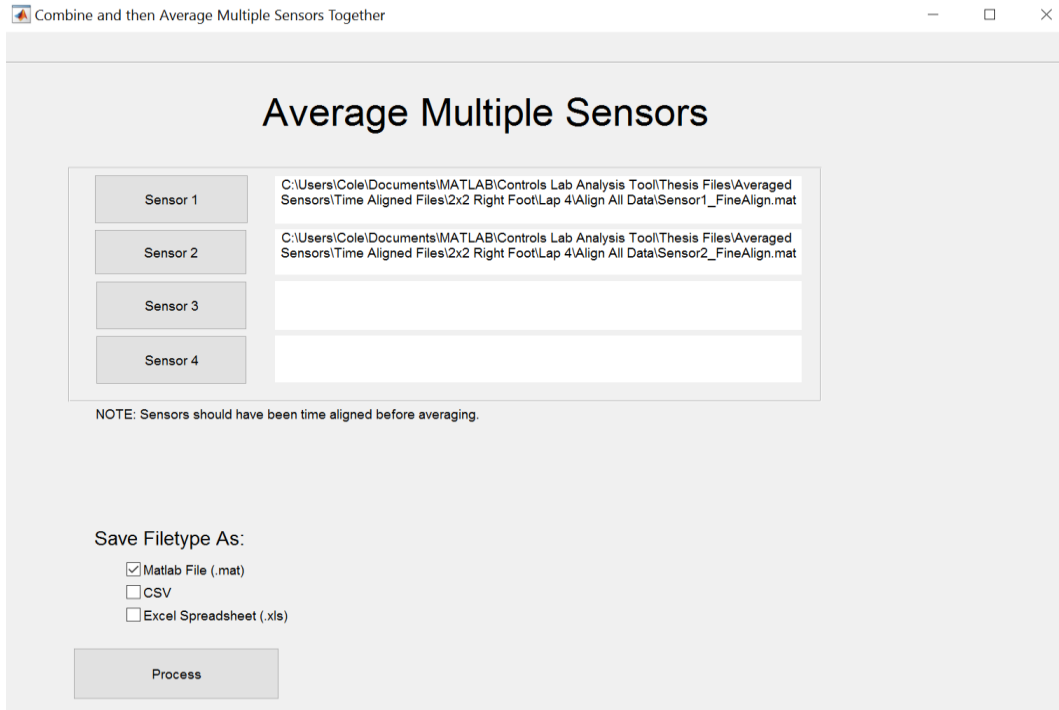


Figure 57. GUI to Fuse and Average Multiple Sensors into One Output

The first sensor modules averaged together were Sensors 3 and 4. Afterward, the only adjustment made to complementary filter settings was in the dynamic-gain value, whose optimal setting was revealed to be  $k_d = 0.01$ . This is a modest change compared to the individually tuned (i.e., not combined) Sensor 3 and 4 settings, which had values of 0.01175 and 0.091, respectively.

The combined and averaged measurements were processed through the PNS. The resulting step-plot, displayed in Figure 58, is a comparison of the two-sensor module averaged performance with that of a single IMU. The single IMU had a step-plot error of 3.83 m (i.e., 0.95% error) while the two-IMU average had a lower error of 2.49 m (i.e., 0.6% error).

Combining two sensor modules and averaging their data slightly improves the overall shape and smoothness of the step-plot. Note that 200 m into the test, the step-plot path representing an individual sensor shows a sudden position error. The two-IMU

averaged sensor path does not have this error. This is due to the inherent benefit achieved by averaging two sensors together; the amplitudes of large errors are reduced.

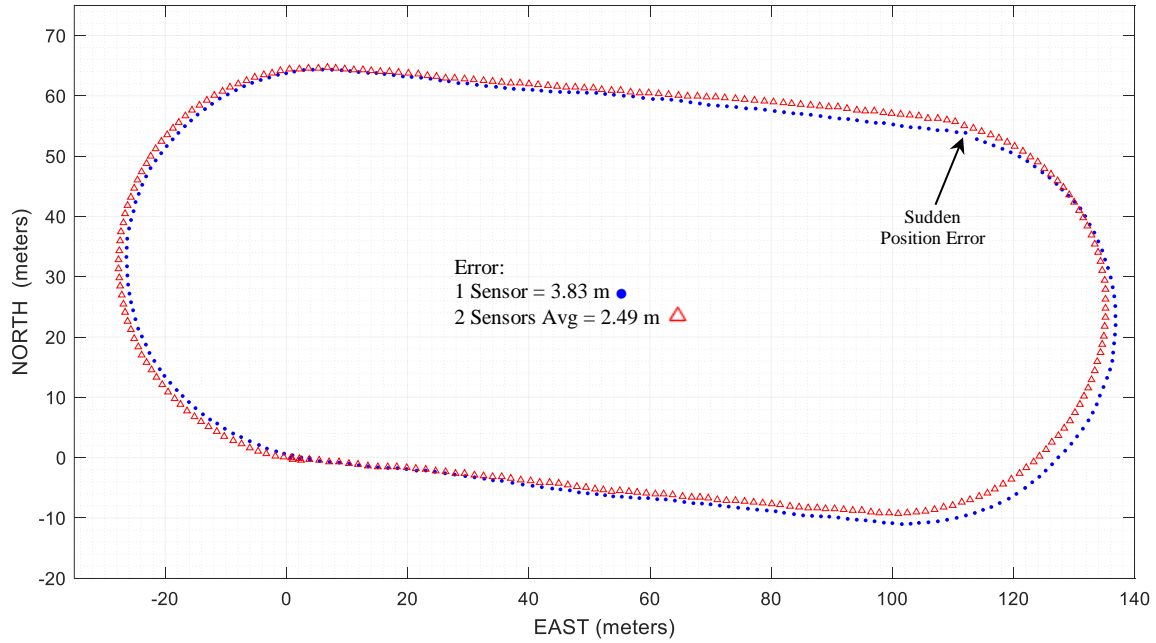


Figure 58. Step-Plot Comparing Single Sensor versus Two Averaged Sensors

### 3. Averaging Three Sensors

Sensors 2, 3, and 4 were combined and averaged, providing a noticeable improvement in accuracy greater than that seen with a two-IMU average, as displayed in Figure 59. With a third sensor module added, the error between the starting and stopping locations was reduced to 0.40 m (i.e., 0.1% error). To achieve such a small error, the dynamic gain  $k_d$  was manually changed to 0.02. Overall smoothness and perceived accuracy are improved with the combination of three IMUs.

A third sensor increases the amount of space needed to be reserved on top of the foot since the  $2 \times 2$  bracket is employed, as well as an accompanied increase in weight. The 3-Space IMUs are small, but when three of them are placed together, their physical dimensions are no longer negligible to the same degree that having one or two IMUs was.

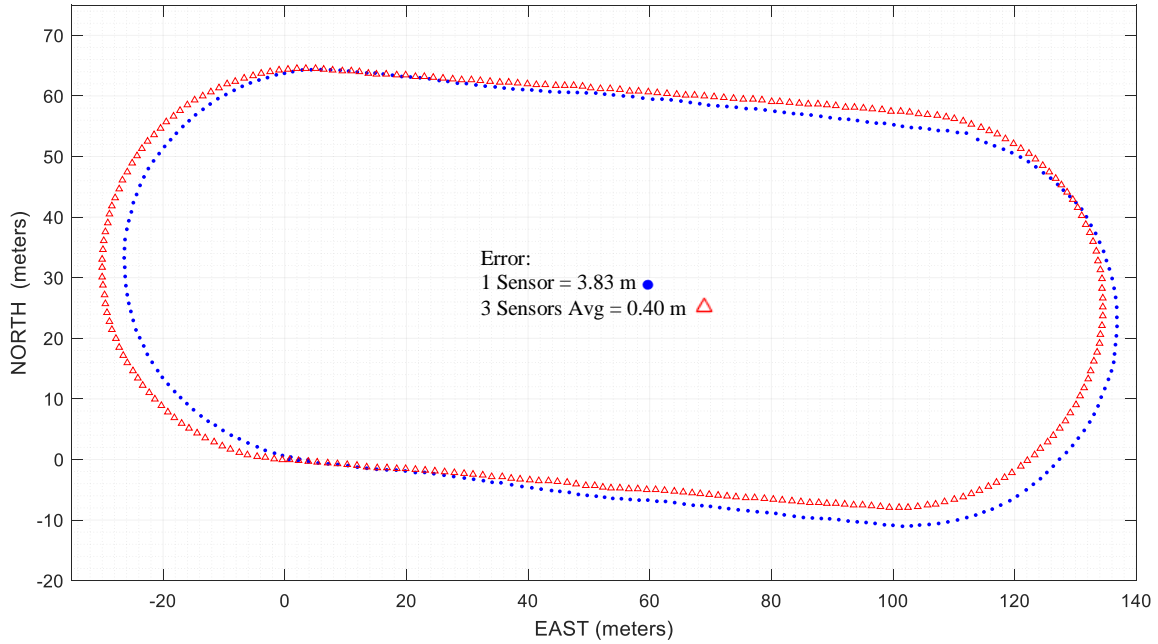


Figure 59. Step-Plot Comparing One Sensor versus Three Averaged Sensors

#### 4. Averaging Four Sensors

Averaging four IMUs together required a dynamic gain  $k_d$  change to 0.04, which suggests the addition of Sensor 4 to the combined average precipitates a more significant adjustment to  $k_d$  than did previous combinations. The resulting plot is presented in Figure 60.

Error increased to a value of 1.56 m (i.e., 0.4% error), worse than the 0.40 m error of three IMUs, which suggests there are disadvantages of averaging multiple sensors together. The system as a whole achieves better accuracy only if each additional sensor equals or exceeds the performance of the others. Compared to averaging IMUs together, special optimal estimation techniques, such as Kalman filtering, may more effectively reduce the negative impacts of a sensor module with large errors. This is because when averaging IMUs together, every sensor has an equal weighting, but optimal estimators adjust each measurement's weight dynamically based on its perceived merit.

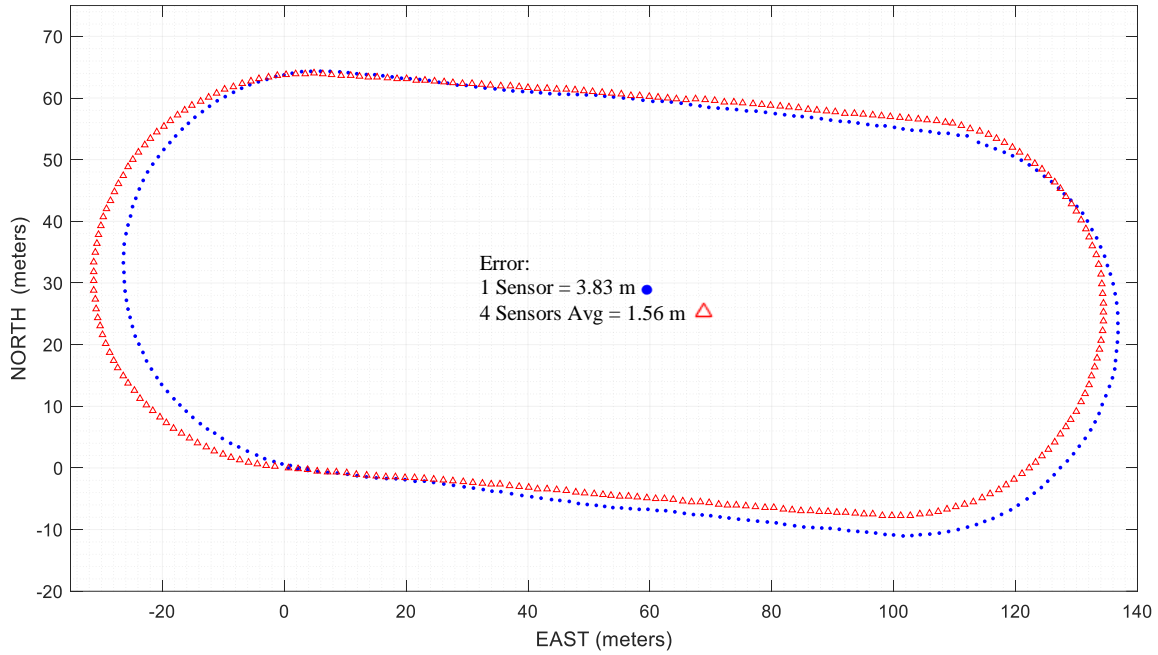


Figure 60. Step-Plot Comparing One Sensor versus Four Averaged Sensors

## 5. Comparison of Additional Averaged Sensors

Data from the previous step-plots are presented in Figures 61–63 to illustrate the benefits of combining different numbers of IMUs. It appears that the greatest accuracy and smoothness improvements occur when going from a single IMU to two averaged IMUs.

The plotted data from a single sensor has a noisier path than the averaged data sets. The random variations typically seen in a single IMU seem to be attenuated by averaging more sensors. Since error from an individual IMU appears random, combining two IMUs will most likely not increase error but attenuate it. The largest magnitude of error for a single measurement, by definition, is reduced when averaging it with a lower error-magnitude value. Assuming that each IMU has exactly the same specifications, if we increase the number of IMUs in the averaging pool of data, the magnitude of errors is reduced. This is why printing several identical sensors onto the same MEMS device is so promising.

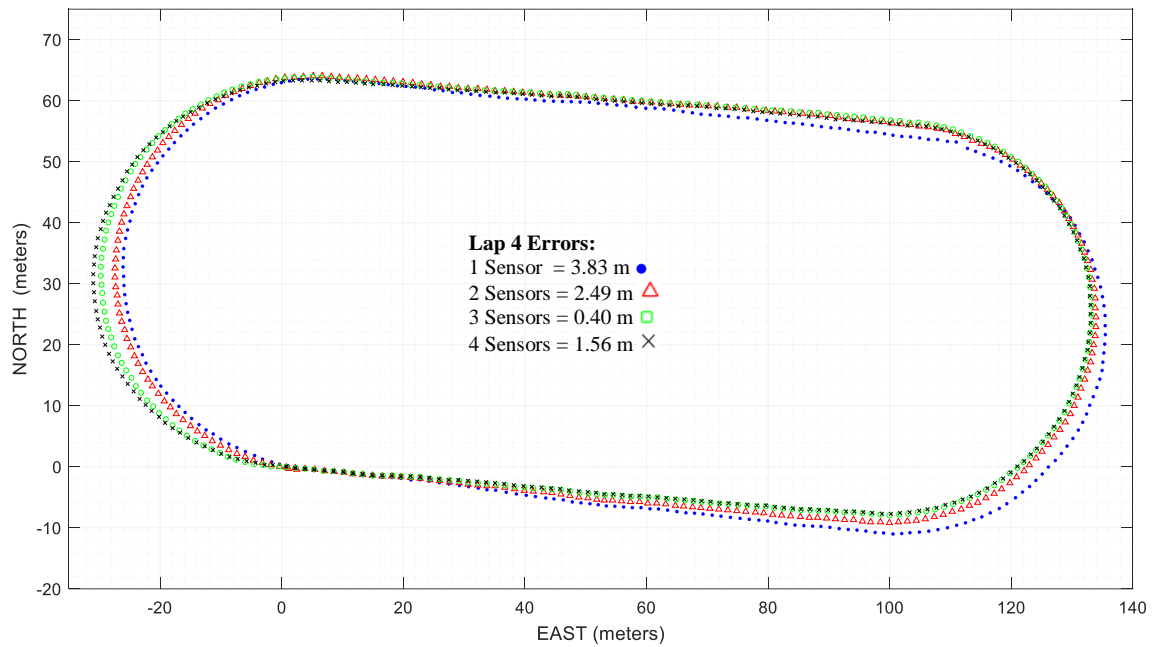


Figure 61. Step-Plot Comparing Increasing Numbers of Averaged Sensors

The plot shown in Figure 62 contains the same data as Figure 61, except the style is altered to better illustrate the performance differences when averaging multiple IMUs.

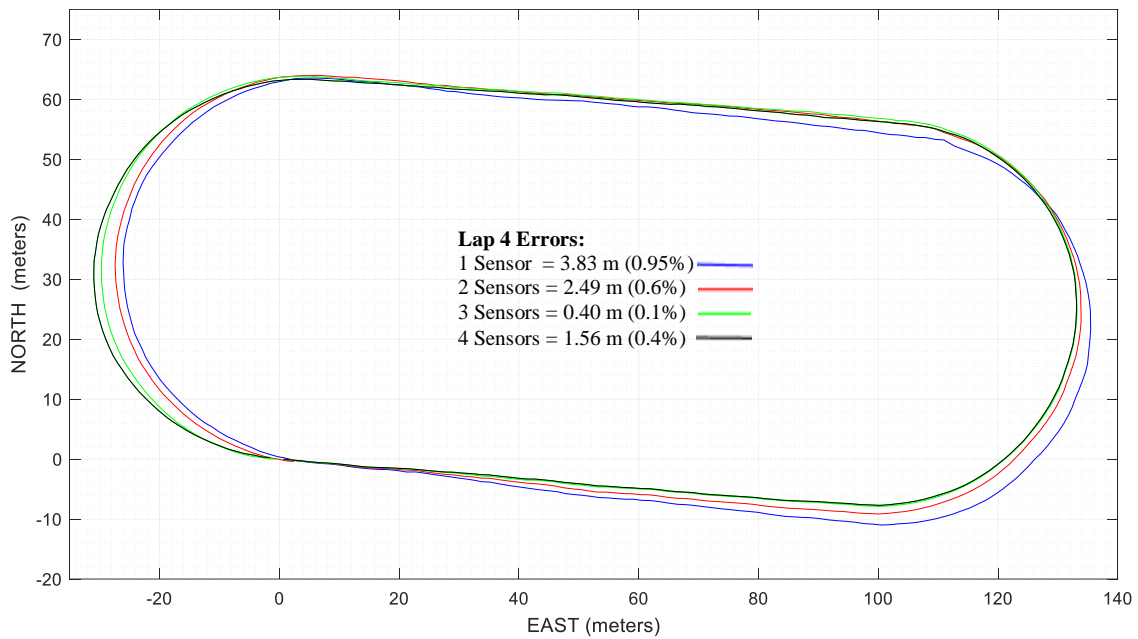


Figure 62. Line-plot Comparing Increasing Numbers of Averaged Sensors

The most significant benefit occurs in going from a single IMU to two averaged IMUs. A smaller difference is observed in adding a third sensor module, while adding a fourth makes little difference to the end result. This type of diminishing return is expected because as the number of IMUs averaged together increases, an individual sensor's ability to improve or degrade performance decreases.

A zoomed-in step-plot is displayed in Figure 63. As in Figure 62, connecting lines are used in place of markers to make it easier to understand. The light-green box located at [0, 0] containing the words “Start/Finish” is where the tester started and ended the test run. The data lines initially appear about 0.7 m to the right of [0, 0] when the right foot impacts the ground during the first step of the lap.

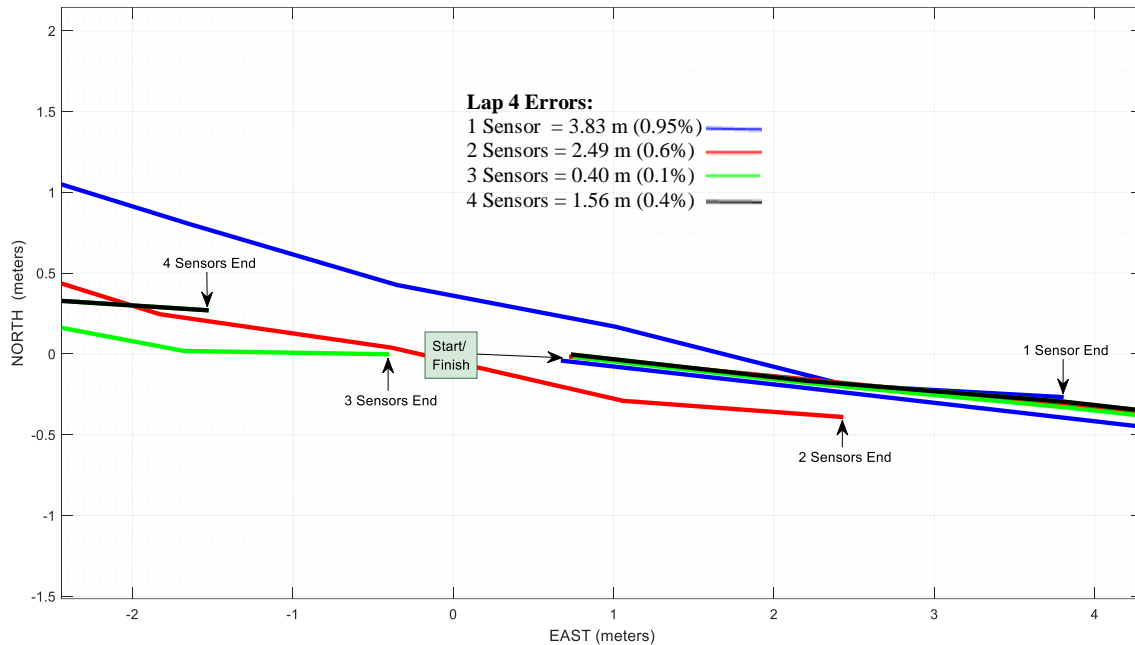


Figure 63. Zoomed-in Step-Plot: Start Versus End Position of Increasing Averaged Sensors; Same Data as Figures 61 and 62

The plot in Figure 63 shows that the overall accuracy of the PNS was good. With a single IMU, an error value near 1% was experienced, while the best accuracy was achieved when three IMUs were averaged together, which had an error of only 0.1%. Of course, without a truth source providing accurate TSPI, a comparison of the position

estimates made by the PNS and the true path cannot be made with a high degree of confidence. That is why an athletic track with a known shape and length was used in testing as a way to effectively compare multiple IMU arrangements with one another. For the case discussed in this section, a knowledge of relative performance is more advantageous than absolute.

The specific complementary filter settings used to achieve the best accuracy with averaged IMUs are presented in Table 8. Every parameter except dynamic gain  $k_d$  is identical; this is mainly because all measurements were derived from foot-mounted sensors. The chosen dynamic-gain value was used to align the step-plot ending locations with respect to the starting location. Changing that value did not adjust the scaling of the step-plot but adjusted heading accuracies.

Table 8. Settings Used to Optimize Performance from Averaged Sensors

Mounting Location: Right Foot	1 Sensor (Sensor 3)	2 Sensors (Sensor 3 & 4)	3 Sensors (Sensors 2, 3, 4)	4 Sensors (Sensor 1, 2, 3, 4)
Gait-Phase Detection Angular-Rate Threshold (deg/s)	0.35	0.35	0.35	0.35
Samples Above/Below Threshold Required to Change Gait-Phase	5	5	5	5
Samples to Save	20	20	20	20
Complementary Filter Angular-Rate Threshold (deg/s)	0.35	0.35	0.35	0.35
$k_s$	1.0	1.0	1.0	1.0
$k_d$	0.026	0.01	0.02	0.04

## **6. Conclusion: More Sensors Enable Better Performance**

Incorporating multiple, co-located IMUs provides a significant improvement in overall PNS accuracy and reduces the sensitivity of the complementary filter to minor setting adjustments. The largest improvement occurs when progressing from one IMU to two. Averaging more than two IMUs together increases accuracy but with diminishing returns as the number of sensors increase.

To achieve the highest-possible performance improvement, it is recommended that two co-located IMUs be mounted on the same foot, secured so as to lock their orientations with respect to one another. Averaging additional sensors is beneficial but presents drawbacks such as greater weight, cost, and complexity for a diminishing gain in performance.

## IV. CONCLUSION

In this chapter, the experimental outcomes are summarized. There were some unexpected results, most notably that distributing sensors over the body does not improve PNS performance. Other key findings are as follows:

- The optimal mounting location of a MEMS IMU for a PNS is the foot.
- Mounting every type of sensor (i.e., Gyroscope, Accelerometer, and Magnetometer) on the foot gives best results.
- Sensors distributed throughout the body degrades performance.
- Averaging together multiple, co-located IMUs improves performance.
- Robust accuracy is predicted to be 10–20 years away, assuming the current rate of development.

### A. OPTIMAL MOUNTING LOCATION: THE FOOT

The foot, shin, lower back, and chest were investigated as mounting locations; the foot provides best performance.

#### 1. Advantages

The greatest forces during walking are exerted at the feet. A foot-mounted IMU allows the gyroscope to take measurements with a high SNR, which provides superb gait-phase detection, effective ZUPTs, and best accuracy. Testing refutes the hypothesis that IMU performance unacceptably degrades under the impact forces of a footstep. Any performance degradation pales in comparison to the inferior performance of alternative mounting locations.

#### 2. Disadvantages

Mounting an IMU to the foot results in small scaling errors. The lower-instep position (i.e., top of foot) typically yields consistently calculated total-path lengths of 396 to 400 m, rather than the true value of 404.6 m. This scaling error is rectified by

multiplying the position output by a value of 1.02. The need for a multiplication factor might be eliminated entirely by mounting the IMU beneath the foot, perhaps in the heel of a boot. This would increase the distance of the sensors from the hip, increasing experienced forces and enabling a more accurate position change estimate for each stride.

Mounting all sensor types on the foot is ideal. PNS performance decreased as the mounting location moved away from the foot because the PNS requires gait-phase detection to determine a stance phase, in which the IMU is assumed perfectly still, to allow the use of ZUPTs to eliminate gyroscope errors. As the mounting location moves away from the foot, gait-phase detection and subsequent ZUPT accuracy degrades. For optimal performance from a single 9-DOF IMU, foot mounting is strongly recommended.

## **B. DISTRIBUTED SYSTEM OF IMUS NOT ADVISED**

Distributing different types of sensors to non-foot locations does not enhance performance, it degrades it. Taking accelerometer and magnetometer data from the lower back and gyroscope from the foot, for example, yields worse performance than seen from a single, combined sensor module on the foot.

Several additional observations were made. PNS performance is not influenced by the location of the magnetometer. Identical performance is found by mounting the gyroscope and accelerometer on the foot while the magnetometer was switched between the lower back and foot. As long as the gyroscope and accelerometer are mounted at the same location—ideally, the foot—the magnetometer may be attached to the lower back or foot without a difference in performance. In addition, PNS performance is poor when the mounting locations of the gyroscope and accelerometer are split up, satisfactory when they are mounted together, and best when they are on the foot.

Finally, considerable effort was given to aligning measurements from different IMUs. Though the IMUs were identical in design, had the same sample rate settings, and were time synchronized just before testing, their sample-rate characteristics showed variation, each having a different timing jitter. Tests began with precise alignment but ended in some degree of deviation. If corrective manual alignment of measurements had not been performed, the data would have yielded distorted outputs in processing.

To achieve the best possible performance from a PNS, it is recommended that only IMU data from the foot be processed. Moreover, splitting mounting locations of the gyroscope and accelerometer results in worse performance than housing them in a single IMU.

### **C. AVERAGING MULTIPLE, CO-LOCATED IMUS IMPROVES PERFORMANCE**

Combining and averaging co-located sensor modules improves PNS performance. The largest improvement occurs when going from a single IMU to two; averaging more than two offers some improvement but with diminishing returns.

Synchronizing multiple co-located IMUs requires a painstaking two-step process of software and manual alignment of measured forces during testing. The first is a rough software alignment that synchronized IMUs in time. The second is accomplished by going into the raw data and manually performing a fine alignment by visually synchronizing experienced forces. This onerous procedure is unacceptable in the field; therefore, it would be desirable to have an automatic, real-time method which can effectively combine sensor data without the use of an RF datalink. A possible solution would be to manufacture MEMS IMUs with multiple identical sensors on the same module. This would allow measurement sets to be synchronized and provide the benefits of averaging multiple co-located IMUs.

### **D. DEVELOPMENT FORECAST**

In this research, a low-cost 3-Space sensor module provided performance similar to, if not better, than that of a considerably more expensive IMU manufactured ten years prior. Typical position errors were less than 1% of the total distance traveled. Nevertheless, there is much room for improvement. To achieve adequate performance, significant time was spent tuning the complementary filter in post-processing, and every IMU required custom tuning. A less intensive customization was required for the same IMU on different test runs. The 3-Space module's large variability between test laps could be reduced by using an IMU with better specifications, specifically in terms of gyroscope-bias stability and ARW.

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX A: PARTIAL DATA ANALYSIS

The following MATLAB algorithms are a small subset of all data processing programs. A method to import and convert data into a MATLAB file type from the YEI TXT files is provided. In addition, the algorithms used to create PSD plots and histograms of sensor noise are shown.

### A. IMPORT LARGE YEI 3-SPACE TEXT FILES INTO MATLAB

Cole Johnson Naval Postgraduate School Monterey, CA 2/03/2016

%%%

clear all

close all

clc

%%%

tic

% This code was originally written by Adam Foushee and adapted by Cole Johnson

nameOfFile= 'EnterFilename\_NoExtension';

Sensor1=importdata([nameOfFile,'.txt']);

Sensor1.textdata=Sensor1.textdata(2:end,:); %Gets rid of header lines

for aa=1:length(Sensor1.textdata(:,2))

    first1=Sensor1.textdata(aa,1);

    second1 = Sensor1.textdata(aa,2);

    third1=Sensor1.textdata(aa,3);

    forth1=Sensor1.textdata(aa,4);

    fifth1=Sensor1.textdata(aa,5);

    hour1=first1{1}(10:11);

    minute1=first1{1}(13:14);

    sec1=first1{1}(16:24);

    hour11(aa)=str2num(hour1);

    minute11(aa)=str2num(minute1);

    sec11(aa)=str2num(sec1);

    time11(aa)=hour11(aa)\*3600+minute11(aa)\*60+sec11(aa);

    gyrox1 = first1{1}((end-7):end);

    gyroy1 = second1{1}(:,:);

    gyroz1 = third1{1}(1:8);

    accel1x=third1{1}((end-7):end);

    accel1y=forth1{1}(1:end);

    accel1z=fifth1{1}(1:8);

    compx1=fifth1{1}((end-7):end);

    gyrox11(aa)=str2num(gyrox1);

    gyroy11(aa)=str2num(gyroy1);

```

gyroz11(aa)=str2num(gyroz1);
accelx11(aa)=str2num(accel1x); % x-acceleration
accely11(aa)=str2num(accel1y); % y-acceleration
accelz11(aa)=str2num(accel1z); % z-acceleration
comp11(aa)=str2num(comp1); % x-Magnetometer reading
end
comp11=Sensor1.data(:,1); % y-Magnetometer reading
compz11=Sensor1.data(:,2); % z-Magnetometer reading

% Places Variables into Structure
% Note: clear command used due to early problems with MATLAB running out of RAM
clearvars -except gyro11 gyro11 gyro11 accelx11 accely11 accelz11 comp11 comp11 compz11 time11 length_comp
Sensor1.comp1 = comp11';
clear comp11
Sensor1.comp1 = comp11;
clear comp11
Sensor1.compz = compz11;
clear compz11
clear length_comp
Sensor1.gyro1 = gyro11';
clear gyro11
Sensor1.gyro1 = gyro11;
clear gyro11
Sensor1.gyro1 = gyro11;
clear gyro11
Sensor1.accel1 = accelx11';
clear accelx11
Sensor1.accel1 = accely11';
clear accely11
Sensor1.accel1 = accelz11';
clear accelz11
Sensor1.time = time11';
clear time11

% Change 'Sensor_'(first column below) to filename you want it saved to
save('Sensor_', 'Sensor1')

toc

```

## B. PSD OF ONE SENSOR

Cole Johnson Naval Postgraduate School Monterey, CA 2/13/2016

%%%

clear all

close all

```
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

## 1. Running of One Sensor

```
load('Sensor1_KFmode_Rawdata')
sensor_num = 'Sensor 1 - '; %Enter Sensor number you want to appear on title
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

## 2. PSDs of Sensors

```
cut_beg = 500; % Number of Samples to cut
cut_end = 500;
% Sample Frequency
Sensor1.time = Sensor1.time(cut_beg :end-cut_end);
fs = 1/mean(diff(Sensor1.time)); %Average Sample Frequency (F_s)
```

## 3. Gyro: PSD

```
Sensor1.gyrox = Sensor1.gyrox(cut_beg :end-cut_end);
Sensor1.gyroy = Sensor1.gyroy(cut_beg :end-cut_end);
Sensor1.gyroz = Sensor1.gyroz(cut_beg :end-cut_end);
figure('Name','Power Spectral Density - Gyro')
pwelch(Sensor1.gyrox,triang(500),250,1024,fs) % Use [] to set parameters to default value
title([sensor_num,'PSD of Gyro X-axis'])
hold on
pwelch(Sensor1.gyroy,triang(500),250,1024,fs)
title([sensor_num,'PSD of Gyro XY-axis'])
pwelch(Sensor1.gyroz,triang(500),250,1024,fs)
title([sensor_num,'PSD of Gyro XYZ-axis'])
%Statistics - Gyro
mean_gyro = mean([Sensor1.gyrox Sensor1.gyroy Sensor1.gyroz]);
std_gyro = std([Sensor1.gyrox Sensor1.gyroy Sensor1.gyroz]);
mean_gyro_all = mean(mean_gyro);
std_gyro_all = mean(std_gyro);
samples_gyro = length(Sensor1.gyrox);
% Number of Samples
total_time = (Sensor1.time(end) - Sensor1.time(1))/(60*60); % Time in hours
total_time_hr = floor(total_time); %Just the num of hrs
total_time_min = round((total_time - total_time_hr)*60);
% Annotations
annotation('textbox',[.15 .8 .1 .1] , 'string', ['\mu = ' num2str(mean_gyro_all,'%0.6f') ' ' '\sigma = ' num2str(std_gyro_all,'%0.6f') "])
annotation('textbox',[.55 .75 .33 .15] , 'string', ['Number Samples = ' num2str(samples_gyro,'%0.0f') ' ' 'Total Time = '
num2str(total_time_hr,'%0.0f') 'hr ' num2str(total_time_min,'%0.0f') 'min ' 'f_{s} = ' num2str(round(fs), '%0.0f') 'Hz'])
```

```
hold off
ylim([-75 -25])
```

## 4. Accelerometer: PSD

```
Sensor1.accelx = Sensor1.accelx(cut_beg :end-cut_end);
Sensor1.accely = Sensor1.accely(cut_beg :end-cut_end);
Sensor1.accelz = Sensor1.accelz(cut_beg :end-cut_end);
% Plot PSD
figure('Name','Power Spectral Density - Accelerometer')
pwelch(Sensor1.accelx,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Accelerometer X-axis'])
hold on
pwelch(Sensor1.accely,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Accelerometer XY-axis'])
pwelch(Sensor1.accelz,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Accelerometer XYZ-axis'])
%Statistics - Accelerometer
mean_accel = mean([Sensor1.accelx Sensor1.accely Sensor1.accelz]);
std_accel = std([Sensor1.gyrox Sensor1.gyroy Sensor1.gyroz]);
mean_accel_all = mean(mean_accel);
std_accel_all = mean(std_accel);
samples_accel = length(Sensor1.accelx);
% Annotations
annotation('textbox',[.15 .8 .1 .1] , 'string', ['\mu = ' num2str(mean_accel_all,'%0.6f') ' ' ; '\sigma = ' num2str(std_accel_all,'%0.6f') ""])
annotation('textbox',[.55 .75 .33 .15] , 'string', ['Number Samples = ' num2str(samples_accel,'%0.0f') ' ' 'Total Time = '
num2str(total_time_hr,'%0.0f') 'hr ' num2str(total_time_min,'%0.0f') 'min ' 'f_{s} = ' num2str(round(fs), '%0.0f') 'Hz'])
hold off
```

## 5. Magnetometer: PSD

```
Sensor1.compx = Sensor1.compx(cut_beg :end-cut_end);
Sensor1.compy = Sensor1.compy(cut_beg :end-cut_end);
Sensor1.compz = Sensor1.compz(cut_beg :end-cut_end);
% Plot PSD
figure('Name','Power Spectral Density - Magnetometer')
pwelch(Sensor1.compx,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Magnetometer X-axis'])
hold on
pwelch(Sensor1.compy,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Magnetometer XY-axis'])
pwelch(Sensor1.compz,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Magnetometer XYZ-axis'])
%Statistics - Magnetometer
mean_comp = mean([Sensor1.compx Sensor1.compy Sensor1.compz]);
```

```

std_comp = std([Sensor1.compx Sensor1.compy Sensor1.compz]);
mean_comp_all = mean(mean_comp);
std_comp_all = mean(std_comp);
samples_comp = length(Sensor1.compx);
% Annotations
annotation('textbox',[.15 .8 .1 .1] , 'string', ['\mu = ' num2str(mean_comp_all,'%0.4f') ' ' ; '\sigma = ' num2str(std_comp_all,'%0.4f') ''])
annotation('textbox',[.55 .75 .33 .15] , 'string', ['Number Samples = ' num2str(samples_comp,'%0.0f') ' ' 'Total Time = '
num2str(total_time_hr,'%0.0f') 'hr ' num2str(total_time_min,'%0.0f') 'min ' 'f_{s} = ' num2str(round(fs), '%0.0f') 'Hz'])
hold off

```

## C. PSDS OF ALL SENSORS

Cole Johnson Naval Postgraduate School Monterey, CA 2/13/2016

%%%

clear all

close all

clc

%%%

### 1. Run All Data

sensor\_num = 'All 4 Sensors - '; %Enter Sensor number you want to appear on title

### 2. Cut Data

cut\_beg = 500; % Number of Samples to cut

cut\_end = 500;

%%%

for i=1:4

if i==1

load('Sensor1\_KFmode\_Rawdata')

elseif i==2

load('Sensor2\_KFmode\_Rawdata')

elseif i==3

load('Sensor3\_KFmode\_Rawdata')

elseif i==4

load('Sensor4\_KFmode\_Rawdata')

end

### 3. PSDs of Sensors

Sensor1.time = Sensor1.time(cut\_beg :end-cut\_end);

fs = 1/mean(diff(Sensor1.time)); %Average Sample Frequency (F\_s)

## 4. Gyro: PSD

```
Sensor1.gyrox = Sensor1.gyrox(cut_beg :end-cut_end);
Sensor1.gyroy = Sensor1.gyroy(cut_beg :end-cut_end);
Sensor1.gyroz = Sensor1.gyroz(cut_beg :end-cut_end);
figure(1)
pwelch(Sensor1.gyrox,triang(500),250,1024,fs) % Use [] to set parameters to default value
title([sensor_num,'PSD of Gyro X-axis'])
hold on
pwelch(Sensor1.gyroy,triang(500),250,1024,fs)
title([sensor_num,'PSD of Gyro XY-axis'])
pwelch(Sensor1.gyroz,triang(500),250,1024,fs)
title([sensor_num,'PSD of Gyro XYZ-axis'])
%Statistics - Gyro
mean_gyro = mean([Sensor1.gyrox Sensor1.gyroy Sensor1.gyroz]);
std_gyro = std([Sensor1.gyrox Sensor1.gyroy Sensor1.gyroz]);
mean_gyro_all = mean(mean_gyro);
std_gyro_all = mean(std_gyro);
samples_gyro = length(Sensor1.gyrox);
% Number of Samples
total_time = (Sensor1.time(end) - Sensor1.time(1))/(60*60); % Time in hours
total_time_hr = floor(total_time); %Just the num of hrs
total_time_min = round((total_time - total_time_hr)*60);
% Annotations
annotation('textbox',[.15 .8 .1 .1] , 'string', ['\mu = ' num2str(mean_gyro_all,'%0.6f') ' ' '\sigma = ' num2str(std_gyro_all,'%0.6f') "])
annotation('textbox',[.55 .75 .33 .15] , 'string', ['Number Samples = ' num2str(samples_gyro,'%0.0f') ' ' 'Total Time = '
num2str(total_time_hr,'%0.0f') 'hr ' num2str(total_time_min,'%0.0f') 'min ' 'f_{s} = ' num2str(round(fs), '%0.0f') 'Hz'])

end
hold off
ylim([-75 -25])
```

## 5. Accelerometer: PSD

```
for i=1:4
if i==1
load('Sensor1_KFmode_Rawdata')
elseif i==2
load('Sensor2_KFmode_Rawdata')
elseif i==3
load('Sensor3_KFmode_Rawdata')
elseif i==4
load('Sensor4_KFmode_Rawdata')
end
Sensor1.time = Sensor1.time(cut_beg :end-cut_end);
fs = 1/mean(diff(Sensor1.time)); %Average Sample Frequency (F_s)
```

```

% Cut off ends to remove effects of button pressing (record on, record off)
Sensor1.accelx = Sensor1.accelx(cut_beg :end-cut_end);
Sensor1.accely = Sensor1.accely(cut_beg :end-cut_end);
Sensor1.accelz = Sensor1.accelz(cut_beg :end-cut_end);
% Plot PSD
figure(2)
pwelch(Sensor1.accelx,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Accelerometer X-axis'])
hold on
pwelch(Sensor1.accely,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Accelerometer XY-axis'])
pwelch(Sensor1.accelz,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Accelerometer XYZ-axis'])
%Statistics - Accelerometer
mean_accel = mean([Sensor1.accelx Sensor1.accely Sensor1.accelz]);
std_accel = std([Sensor1.gyrox Sensor1.gyroy Sensor1.gyroz]);
mean_accel_all = mean(mean_accel);
std_accel_all = mean(std_accel);
samples_accel = length(Sensor1.accelx);
% Annotations
annotation('textbox',[.15 .8 .1 .1] , 'string', ['\mu = ' num2str(mean_accel_all,'%0.6f') ' ' ; '\sigma = ' num2str(std_accel_all,'%0.6f') ''])
annotation('textbox',[.55 .75 .33 .15] , 'string', ['Number Samples = ' num2str(samples_accel,'%0.0f') ' ' 'Total Time = '
num2str(total_time_hr,'%0.0f') 'hr ' num2str(total_time_min,'%0.0f') 'min ' 'f_{s} = ' num2str(round(fs), '%0.0f') 'Hz'])
end
hold off

```

## 6. Magnetometer: PSD

```

for i=1:4
if i==1
load('Sensor1_KFmode_Rawdata')
elseif i==2
load('Sensor2_KFmode_Rawdata')
elseif i==3
load('Sensor3_KFmode_Rawdata')
elseif i==4
load('Sensor4_KFmode_Rawdata')
end
Sensor1.time = Sensor1.time(cut_beg :end-cut_end);
fs = 1/mean(diff(Sensor1.time)); %Average Sample Frequency (F_s)
% Cut off ends to remove effects of button pressing (record on, record off)
Sensor1.compx = Sensor1.compx(cut_beg :end-cut_end);
Sensor1.compy = Sensor1.compy(cut_beg :end-cut_end);
Sensor1.compz = Sensor1.compz(cut_beg :end-cut_end);
% Plot PSD
figure(3)

```

```

pwelch(Sensor1.compx,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Magnetometer X-axis'])
hold on
pwelch(Sensor1.compy,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Magnetometer XY-axis'])
pwelch(Sensor1.compz,triang(500),250,1024,fs)
title([sensor_num, 'PSD of Magnetometer XYZ-axis'])

%Statistics - Magnetometer
mean_comp = mean([Sensor1.compx Sensor1.compy Sensor1.compz]);
std_comp = std([Sensor1.compx Sensor1.compy Sensor1.compz]);
mean_comp_all = mean(mean_comp);
std_comp_all = mean(std_comp);
samples_comp = length(Sensor1.compx);

% Annotations
annotation('textbox',[.15 .8 .1 .1] , 'string', ['\mu = ' num2str(mean_comp_all,'%0.4f') ' ' ; '\sigma = ' num2str(std_comp_all,'%0.4f') "])
annotation('textbox',[.55 .75 .33 .15] , 'string', ['Number Samples = ' num2str(samples_comp,'%0.0f') ' ' 'Total Time = '
num2str(total_time_hr,'%0.0f') 'hr ' num2str(total_time_min,'%0.0f') 'min ' 'f_{s} = ' num2str(round(fs), '%0.0f') 'Hz'])
end
hold off

```

## D. HISTOGRAM PLOT OF SENSOR

### 1. Plot Noise Histogram of One Sensor

Cole Johnson Naval Postgraduate School Monterey, CA 2/13/2016

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

### 2. Run One Sensor

```

load('Sensor1_KFmode_Rawdata')
sensor_num = 'Sensor 1 - ' ; %Enter Sensor number you want to appear on title

```

### 3. Cut Data

```

cut_beg = 500;
cut_end = 500;

```

### 4. Histogram

```
noise_histogram_func(Sensor1.gyrox, Sensor1.gyroy, Sensor1.gyroz, Sensor1.accelx, Sensor1.accelz,
Sensor1.compx, Sensor1.compy, Sensor1.compz, cut_beg, cut_end, sensor_num)
```

## E. HISTOGRAM FUNCTION

### 1. Sensor Noise Characteristics - Histogram

Cole Johnson Naval Postgraduate School Monterey , CA 2/13/2016

```
function [] = noise_histogram_func(gyrox, gyroy, gyroz, accelx, accelz, compx, compy, compz, cut_beg, cut_end, sensor_num)
```

### 2. Histogram for Sensor

```
gyrox = gyrox(cut_beg :end-cut_end);
gyroy = gyroy(cut_beg :end-cut_end);
gyroz = gyroz(cut_beg :end-cut_end);
accelx = accelx(cut_beg :end-cut_end);
accely = accely(cut_beg :end-cut_end);
accelz = accelz(cut_beg :end-cut_end);
compx = compx(cut_beg :end-cut_end);
compy = compy(cut_beg :end-cut_end);
compz = compz(cut_beg :end-cut_end);
```

### 3. Gyro

Find Statistics of Data

```
mean_gx = mean(gyrox)
mean_gy = mean(gyroy)
mean_gz = mean(gyroz)
mean_g = mean([mean_gx mean_gy mean_gz])
std_gx = std(gyrox)
std_gy = std(gyroy)
std_gz = std(gyroz)
std_g = mean([std_gx std_gy std_gz])

% Histogram Settings
bwidth = .0011;
blimits = [-7e-3 7e-3];

% Histogram
figure('Name','Histogram of Sensor Data - Gyro')
h1 = histogram(gyrox,'BinWidth',bwidth,'BinLimits', blimits)
hold on
h2 = histogram(gyroy,'BinWidth',bwidth,'BinLimits', blimits)
```

```

h3 = histogram(gyroz,'BinWidth',bwidth, 'BinLimits', blimits)
title([sensor_num,'Histogram of Gyro XYZ-axis'])
xlabel('Degrees Per Second')
ylabel('Probability')
legend('Gyro 1', 'Gyro 2', 'Gyro 3')

hold off

% Histogram of Norm
bwidth = 0.0011;
gyro_all = [gyrox; gyroy; gyroz];
figure('Name', 'Combined Histogram of Sensor Data - Gyro')
h = histogram(gyro_all,'BinLimits',blimits,'BinWidth',bwidth)
title([sensor_num,'Histogram of Gyro - Combined'])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## 4. Accelerometer

```

mean_ax = mean(accelx)
mean_ay = mean(accely)
mean_az = mean(accelz)
norm_a = norm([mean_ax mean_ay mean_az])
std_ax = std(accelx)
std_ay = std(accely)
std_az = std(accelz)
std_a = mean([std_ax std_ay std_az])

% Histogram Settings
bwidth = .001;
blimits = [-3e-2 3e-2];

% Histogram
figure('Name','Histogram of Sensor Data - Accelerometer')
h1a = histogram(accelx,'BinWidth',bwidth,'BinLimits', blimits)

hold on
h2a = histogram(accely,'BinWidth',bwidth,'BinLimits', blimits)
h3a = histogram(accelz,'BinWidth',bwidth, 'BinLimits', blimits)
title([sensor_num,'Histogram of Accelerometer YZ-axis'])
xlim([0 0.03])
xlabel('g-force')
ylabel('Probability')
legend([h2a h3a],{'Accel 2','Accel 3'})
hold off

hold off

```

```

% Combined Histogram
bwidth = 0.001;
accel_all = [accelx; accely; accelz];
figure('Name', 'Combined Histogram of Sensor Data - Accelerometer')
ha = histogram(accel_all,'BinLimits',blimits,'BinWidth',bwidth)
title([sensor_num,'Histogram of Accelerometer - Combined'])
xlim([0 0.03])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## 5. Magnetometer

```

mean_cx = mean(compx)
mean_cy = mean(compy)
mean_cz = mean(compz)
norm_c = norm([mean_cx mean_cy mean_cz])
std_cx = std(compx)
std_cy = std(compy)
std_cz = std(compz)
std_c = mean([std_cx std_cy std_cz]); %standard deviation of data

```

```

% Histogram Settings
bwidth = .00115;
blimits = [-3e-1 3e-1];
blimits = [.145 .245];
% Histogram
figure('Name','Histogram of Sensor Data - Magnetometer')
h1c = histogram(compx,'BinWidth',bwidth,'BinLimits', blimits)

hold on
h2c = histogram(compy,'BinWidth',bwidth,'BinLimits', blimits)
h3c = histogram(compz,'BinWidth',bwidth, 'BinLimits', blimits)
title([sensor_num,'Histogram of Magnetometer YZ-axis'])
xlabel('Gauss')
ylabel('Probability')
legend([h2c h3c],{'Magnetometer 2','Magnetometer 3'})

```

```

hold off
% Histogram of Norm
bwidth = 0.0004;
comp_all = [compx; compy; compz];
figure('Name', 'Combined Histogram of Sensor Data - Magnetometer')
hc = histogram(comp_all,'BinLimits',blimits,'BinWidth',bwidth)
title([sensor_num,'Histogram of Magnetometer - Combined'])
end

```

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX B: DATA COLLECTION

Three separate data collection test events took place and are listed in Table 9.

Table 9. Data Collection Events

Date of Test	Lap Name	Bracket Used	Number of Steps	Sensor 1 Location	Sensor 2 Location	Sensor 3 Location	Sensor 4 Location
16 Feb. 2016	Lap 1	2×1	280	LF	LF	RF	RF
16 Feb. 2016	Lap 2	2×1	269	LF	LF	RF	RF
16 Feb. 2016	Lap 3	2×1	267	LF	LF	RF	RF
16 Feb. 2016	Lap 4	2×2	266	RF	RF	RF	RF
16 Feb. 2016	Lap 5	2×2	269	RF	RF	RF	RF
16 Feb. 2016	Lap 6	2×2 Inverted Orientation	266	RF	RF	RF	RF
22 Feb. 2016	Lap 7	2×1	269	LB	LB	RF	RF
22 Feb. 2016	Lap 8	2×1	260	Chest	Chest	RF	RF
22 Feb. 2016	Lap 9	2×1	256	LB	LB	RF	RF
11 Sept. 2015	Lap 10	Blue Rubber	N/A	LF	LF	RF	RF
11 Sept. 2015	Lap 11	Blue Rubber	N/A	LF	LF	RF	RF
11 Sept. 2015	Lap 12	Blue Rubber	N/A	LF	LF	RF	RF
11 Sept. 2015	Lap 13	Blue Rubber	N/A	LF	LF	RF	RF
11 Sept. 2015	Lap 14	Blue Rubber	N/A	LF	LF	RF	RF
11 Sept. 2015	Lap 15	Blue Rubber	N/A	LF	LF	RF	RF

\*LF = Left Foot; RF = Right Foot; LB = Lower-Back.

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF REFERENCES

- [1] Federal Aviation Admin. (2016, May 5). Equip ADS-B FAQ [Online]. Available: <http://www.faa.gov/nextgen/equipadsb/faq/#q18>. Accessed May 24, 2016.
- [2] Nat. Coordination Office for Space-Based Positioning, Navigation, and Timing. (2015, Dec. 11). Space Segment [Online]. Available: <http://www.gps.gov/systems/gps/space/>. Accessed Jan. 8, 2016.
- [3] Department of Defense, “Global Positioning System Standard Positioning Service Performance Standard 4th Edition,” Sept. 2008. [Online]. Available: <http://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf>. Accessed Jan. 8, 2016.
- [4] Receiver autonomous integrity monitoring. ( 2015, June 15). *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Receiver\\_autonomous\\_integrity\\_monitoring](https://en.wikipedia.org/wiki/Receiver_autonomous_integrity_monitoring). Accessed Jan. 10, 2016.
- [5] Dilution of precision (GPS). (2016, Jan. 9). *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Dilution\\_of\\_precision\\_\(GPS\)](https://en.wikipedia.org/wiki/Dilution_of_precision_(GPS)). Accessed Jan. 10, 2016.
- [6] X. Yun, J. Calusdian, E. R. Bachmann, and R. B. McGhee, “Estimation of human foot motion during normal walking using inertial and magnetic sensor measurements,” *IEEE Trans. on Instrumentation and Measurement*, vol. 61, no. 7, pp. 2059–2072, July 2012.
- [7] J. Driesslein, “Scalable mobile ad hoc network (MANET) to enhance situational awareness in distributed small unit operations,” M.S. thesis, Dept. Elect. Comput. Eng., Naval Postgraduate School, Monterey, CA, 2015.
- [8] C. Khan, “Geometry-of-fire tracking algorithm for direct-fire weapon systems,” M.S. thesis, Dept. Elect. Comput. Eng., Naval Postgraduate School, Monterey, CA, 2015.
- [9] K. Reese, “A situational-awareness system for networked infantry including an accelerometer-based shot-identification algorithm for direct-fire weapons,” M.S. thesis, Dept. Elect. Comput. Eng., Naval Postgraduate School, Monterey, CA, 2016.
- [10] A. Foushee, “Using posture estimation to enhance personal inertial tracking,” M.S. thesis, Dept. Elect. Comput. Eng., Naval Postgraduate School, Monterey, CA, 2016.

- [11] J. Calusdian, "A personal navigation system based on inertial and magnetic field measurements," Ph.D. dissertation, Dept. Elect. Comput. Eng., Naval Postgraduate School, Monterey, CA, 2010.
- [12] E. W. Nye. Pounds sterling to dollars: historical conversion of currency. (n.d.). [Online]. Available: <http://www.uwyo.edu/numimage/currency.htm>. Accessed Jan. 13, 2016.
- [13] D. Sobel, *Longitude: the True Story of a Lone Genius Who Solved the Greatest Scientific Problem of his Time*, New York City, NY: Walker Publishing Company, Inc., 1995.
- [14] V-2 rocket. (n.d.). *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/V-2\\_rocket#Technical\\_details](https://en.wikipedia.org/wiki/V-2_rocket#Technical_details). Accessed Jan. 13, 2016.
- [15] A. Teitel. (2013, Apr. 22). Project Hermes: America's V-2 rocket program [Online]. Available: <http://news.discovery.com/space/history-of-space/project-hermes-americas-v-2-rocket-program-130422.htm>. Accessed May 18, 2016.
- [16] A. King, "Inertial navigation: Forty years of evolution," *Gen. Elect. Co. Rev.*, vol. 13, no. 3, pp. 140–149, 1998.
- [17] Inertial navigation system. (n.d.). *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Inertial\\_navigation\\_system#Strapdown\\_systems](https://en.wikipedia.org/wiki/Inertial_navigation_system#Strapdown_systems). Accessed Jan. 14, 2016.
- [18] J.-N. Juang and R. Radharamanan, "Evaluation of ring laser and fiber optic gyroscope technology," presented at American Society For Engineering Education Section Conference, Middle Atlantic, 2009.
- [19] Ring Laser. (n.d.). *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Ring\\_laser](https://en.wikipedia.org/wiki/Ring_laser). Accessed Jan. 14, 2016.
- [20] Fibre optic gyroscope. (n.d.). *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Fibre\\_optic\\_gyroscope](https://en.wikipedia.org/wiki/Fibre_optic_gyroscope). Accessed Jan. 15, 2016.
- [21] Nat. Coordination Office for Space-Based Positioning, Navigation, and Timing. (2014, July 30). GPS Modernization [Online]. Available: <http://www.gps.gov/systems/gps/modernization/>. Accessed Jan. 18, 2016.

- [22] W. J. Hughes Tech. Center NSTB/WAAS T&E Team, “Global Positioning System (GPS) Standard Positioning Service (SPS) Performance Analysis Report,” Fed. Aviation Admin., Atlantic City, NJ, Rep. 86, July 2014. [Online]. Available: [http://www.nstb.tc.faa.gov/reports/PAN86\\_0714.pdf](http://www.nstb.tc.faa.gov/reports/PAN86_0714.pdf). Accessed Jan. 18, 2016.
- [23] Nat. Coordination Office for Space-Based Positioning, Navigation, and Timing. (2015, Aug. 27). New Civil Signals [Online]. Available: <http://www.gps.gov/systems/gps/modernization/civilsignals/>. Accessed Jan. 18, 2016.
- [24] D. Askew. (n.d.). MEMS: A Brief Overview [Online]. Available: <http://www.mouser.in/applications/mems-overview/>. Accessed Jan. 25, 2016.
- [25] MEMS. (2015, Sept. 17). *Engineering and Technology History Wiki*. [Online]. Available: <http://ethw.org/MEMS>. Accessed Jan. 18, 2016.
- [26] D. E. Dilger. (2010, Jun. 16). Inside iPhone 4: Gyro spins Apple ahead in gaming [Online]. Available: [http://appleinsider.com/articles/10/06/16/inside\\_iphone\\_4\\_gyro\\_spins\\_apple\\_ahead\\_in\\_gaming/page/2](http://appleinsider.com/articles/10/06/16/inside_iphone_4_gyro_spins_apple_ahead_in_gaming/page/2). Accessed Jan. 18, 2016.
- [27] RobotShop, Search results for ‘IMU’. (2016, Jan. 18). RobotShop.com. [Online]. Available: <http://www.robotshop.com/en/catalogsearch/result/index/?dir=asc&order=price&q=IMU>. Accessed Jan. 18, 2016.
- [28] MicroStrain 3DM-GX4-25. (2015). LORD MicroStrain. [Online]. Available: <http://www.microstrain.com/inertial/3dm-gx4-25>. Accessed Jan. 25, 2016.
- [29] Yost Labs, Portsmouth, OH. (2015). *YEI 3-Space Sensor Software Suite*. [Online]. Available: <https://yostlabs.com/yost-labs-3-space-sensor-software-suite/>. Accessed Aug. 20, 2016.
- [30] MVN Biomech. (n.d.). XSENS [Online]. Available: <https://www.xsens.com/products/mvn-biomech/>. Accessed Jan. 25, 2016.
- [31] Piezoelectric accelerometer. (n.d.) *Wikipedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Piezoelectric\\_accelerometer](https://en.wikipedia.org/wiki/Piezoelectric_accelerometer). Accessed Jan. 16, 2016.
- [32] F. Pasolini. (2011, Apr. 11). MEMS Accelerometers, Gyroscopes, and Geomagnetic Sensors - Propelling Disruptive Consumer Applications. *Apple Insider* [Online]. Available: <http://www.digkey.com/en/articles/techzone/2011/apr/mems-accelerometers-gyroscopes-and-geomagnetic-sensors---propelling-disruptive-consumer-applications>. Accessed Jan. 25, 2016.

- [33] NXP Semiconductor. (2016). MMA8451Q:  $\pm 2g/\pm 4g/\pm 8g$ , Low g, 14-bit digital accelerometer [Online]. Available: <http://www.nxp.com/products/sensors/accelerometers/3-axis-accelerometers/2g-4g-8g-low-g-14-bit-digital-accelerometer:MMA8451Q>. Accessed Jan. 25, 2016.
- [34] G-force. (n.d.). *Wikipedia*. [Online]. Available: <https://en.wikipedia.org/wiki/G-force>. Accessed Feb. 7, 2016.
- [35] Gravity in Monterey, CA. (2016, Feb. 7). WolframAlpha. [Online]. Available: <https://www.wolframalpha.com>. Accessed Feb. 7, 2016.
- [36] Maxim Integrated. (2016). MAX21000 Ultra-Accurate, Low Power, 3-Axis Digital Output Gyroscope [Online]. Available: <https://www.maximintegrated.com/en/products/analog/sensors-and-sensor-interface/MAX21000.html>. Accessed Jan. 25, 2016.
- [37] Honeywell. (2012, Jan.). 3-Axis digital compass IC HMC5983 [Online]. Available: [https://www51.honeywell.com/aero/common/documents/myaerospacecatalog-documents/Defense\\_Brochures-documents/HMC5983\\_3\\_Axis\\_Compass\\_IC.pdf](https://www51.honeywell.com/aero/common/documents/myaerospacecatalog-documents/Defense_Brochures-documents/HMC5983_3_Axis_Compass_IC.pdf). Accessed Jan. 25, 2016.
- [38] Z. Berman, “The design process for navigation Kalman filters: striving for performance and quality,” presented at IEEE/ION Position, Location and Navigation Symposium, Monterey, CA, 2014.
- [39] O. J. Woodman, *An Introduction to Inertial Navigation*, Cambridge, UK: University of Cambridge, 2007.
- [40] M. Grewal and A. Andrews. (2010, Feb.). How Good is Your Gyro?. *IEEE Control Systems Magazine* [Online]. Available: <http://ieeecss.org/CSM/library/2010/feb10/06-AsktheExperts.pdf>. Accessed Jan. 28, 2016.
- [41] W. Stockwell. “Angle random walk,” Crossbow Tech. Inc., Milpitas, CA. Tech. Rep. (2003). Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.210.1133&rep=rep1&type=pdf>. Accessed Aug. 20, 2016.
- [42] *3-Space Sensor Minature Attitude & Heading Reference System User’s Manual*, YEI Tech., Portsmouth, OH, 2014.
- [43] KVH. (2016). 1175 IMU [Online]. Available: <http://www.kvh.com/Military-and-Government/Gyros-and-Inertial-Systems-and-Compasses/Gyros-and-IMUs-and-INS/IMUs/1175-IMU.aspx>. Accessed Jan. 27, 2016.

- [44] MicroStrain. (n.d.). Detailed specifications for 3DM-GX1 [Online]. Available: <http://files.microstrain.com/3DM-GX1%20Detailed%20Specs%20-%20Rev%201%20-%200070723.pdf>. Accessed Jan. 27, 2017.
- [45] XSENS. (2012). MTw development kit leaflet download [Online]. Available: <https://www.xsens.com/wp-content/uploads/2013/11/mtw-leaflet.pdf>. Accessed Jan. 27, 2016.
- [46] D. Arch. (2010, Dec. 2). High-Performance MemS Gyroscopes: Current Status and Emerging Trends [Online]. Available: <http://www.memsjournal.com/2010/12/high-performance-mems-gyroscopes-current-status-and-emerging-trends.html>. Accessed Jan. 27, 2016.
- [47] W. Stockwell. "Bias stability measurement: Allen variance," Crossbow Tech. Inc., Milpitas, CA. Tech. Rep. (2004).
- [48] R. L. Klika. (2008, Jan. 10). Iowa Army National Guard mobilization training [Photograph]. Available: [http://archive.defense.gov/dodcmssshare/photoessay/2008-01/hires\\_080110-A-0559K-168.jpg](http://archive.defense.gov/dodcmssshare/photoessay/2008-01/hires_080110-A-0559K-168.jpg). Accessed Feb. 8, 2016.
- [49] H. Fourati, "Heterogeneous Data Fusion Algorithm for Pedestrian Navigation via Foot-Mounted Inertial Measurement Unit and Complementary Filter," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 1, pp. 221–229, Jan. 2015.
- [50] I. Skog, J. O. Nilsson, and P. Händel, "Evaluation of zero-velocity detectors for foot-mounted inertial navigation systems," *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, Zurich, 2010, pp. 1–6.
- [51] Z. Xiao, H. Wen, A. Markham and N. Trigoni, "Indoor Tracking Using Undirected Graphical Models," in *IEEE Transactions on Mobile Computing*, vol. 14, no. 11, pp. 2286–2301, Nov. 1, 2015.
- [52] M. N. Muhammad, Z. Salcic and K. I. -K. Wang, "Subtractive Clustering as ZUPT Detector," *IEEE Intl Conf on Ubiquitous Intelligence and Computing*, Bali, 2014, pp. 349–355.
- [53] M. Laverne, M. George, D. Lord, A. Kelly, T. Mukherjee, "Experimental Validation of Foot to Foot Range Measurements in Pedestrian Tracking," *Proceedings of the International Technical Meeting of The Satellite Division of the Institute of Navigation*, Portland, OR, 2011, pp. 1386-1393.

- [54] R. Girisha, G. V. Prateek, K. V. S. Hari and P. Händel, “Fusing the navigation information of dual foot-mounted zero-velocity-update-aided inertial navigation systems,” *International Conference on Signal Processing and Communications*, Bangalore, 2014, pp. 1–6.
- [55] J. B. Bancroft and G. Lachapelle, “Data fusion algorithms for multiple inertial measurement units,” *Sensors*, vol. 11, no. 7, pp. 6771–6798, Jun. 2011.
- [56] H. Chang, L. Xue, W. Qin, G. Yuan and W. Yuan, “An integrated MEMS gyroscope array with higher accuracy output,” *Sensors*, vol. 8, no. 4, pp. 2886–2899, Apr. 2008.
- [57] MAT-File Versions. (2016). MathWorks. [Online]. Available: [http://www.mathworks.com/help/matlab/import\\_export/mat-file-versions.html](http://www.mathworks.com/help/matlab/import_export/mat-file-versions.html). Accessed Feb. 9, 2016.
- [58] B. V. Veen, Interviewee. (2013 Jun. 13). The Power Spectral Density. [YouTube video]. Available: <https://www.youtube.com/watch?v=-Nt0FaofWL4>. Accessed Feb. 9, 2016.
- [59] College of Engineering, “Chapter 11: Noise and Noise Rejection,” in *ME365: Measurement Systems*, West Lafayette, IN: Purdue Univ., 2011, pp. 6. Available <https://engineering.purdue.edu/ME365/Textbook/chapter11.pdf>. Accessed Feb. 10, 2016.
- [60] Pwelch. (2016). MathWorks. [Online]. Available: <http://www.mathworks.com/help/signal/ref/pwelch.html>. Accessed Feb. 12, 2016.
- [61] P. Welch, “The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms,” in *IEEE Transactions on Audio and Electroacoustics*, vol. 15, no. 2, pp. 70–73, Jun 1967.
- [62] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, 2nd ed. Hoboken, NJ: Wiley, 2003, p. 52.
- [63] J. B. Kuipers, *Quaternions and Rotation Sequences*, Princeton, NJ: Princeton Univ. Press, 1999.
- [64] CSUMB Athletic Track, Seaside, CA [Google Image], 2015.
- [65] T. Oberg, A. Karsznia and K. Oberg, “Basic gait parameters: reference data for normal subjects, 10–79 years of age,” *Journal of rehabilitation research and development*, vol. 30, no. 2, pp. 210–223, Jan. 1993.

[66] P. Hannon. Hannon Biomechanics Analysis. Email interview. Apr. 7, 2015.

[67] J. Ashworth. Yost Labs. Telephone interview. Jan. 2016.

THIS PAGE INTENTIONALLY LEFT BLANK

## **INITIAL DISTRIBUTION LIST**

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California